**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) Обернути рядок в зворотному порядку.

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

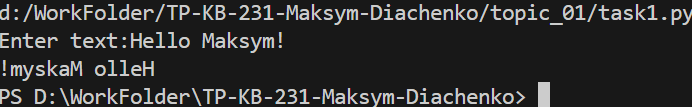
3) Написати функцію пошуку дискримінанту квадратного рівняння.

**Обернення рядка**

Необхідно рядок, наприклад "Hello Maksym!" перетворити наступним чином "!myskaM olleH".

Хід виконання завдання:

У цьому завданні я використав функцію, яка обертає вміст рядка.



Текст програми:

def reverse\_string(text):

    return text[::-1]

text = input("Enter text:")

reversed\_text = reverse\_string(text)

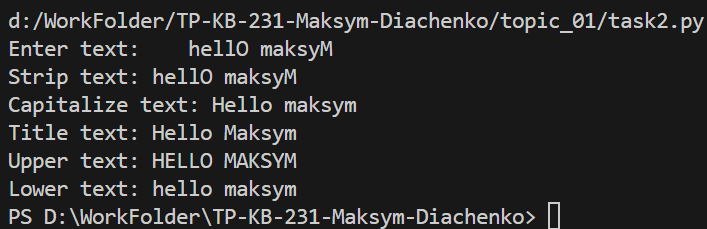
print(reversed\_text)

**Тестування функцій, що працюють з рядками**

Необхідно виконати тестування функцій, що працюють з рядками: strip(), capitalize(), title(), upper(), lower().

Хід виконання завдання:

У цьому завданні я використав вище наведені функції.



Функція Strip прибирає пробіли по краям рядка.

Функція Capitalize робить першу букву першого слова в рядку велику, а всі інші маленькими.

Функція Tittle перетворює регістри перших букв усіх слів рядка великими.

Функція Upper перетворює регістри всіх букв усіх слів рядка великими.

Функція Lower перетворює регістри всіх букв усіх слів рядка маленькими.

Текст програми:

text = input("Enter text:")

text1=text.strip()

print("Strip text: " + text1)

text2=text1.capitalize()

print("Capitalize text: " + text2)

text3=text2.title()

print("Title text: " + text3)

text4=text3.upper()

print("Upper text: " + text4)

text5=text4.lower()

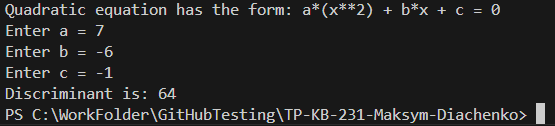
print("Lower text: " + text5)

**Написання функції пошуку дискримінанту квадратного рівняння**

Необхідно написати функцію пошуку дискримінанту квадратного рівняння.

Хід виконання завдання:

У цьому завданні я написав функцію пошуку дискримінанту квадратного рівняння.



Текст програми:

def discr(a, b, c):

    return b\*\*2 - 4\*a\*c

print("Quadratic equation has the form: a\*(x\*\*2) + b\*x + c = 0")

a = int(input("Enter a = "))

b = int(input("Enter b = "))

c = int(input("Enter c = "))

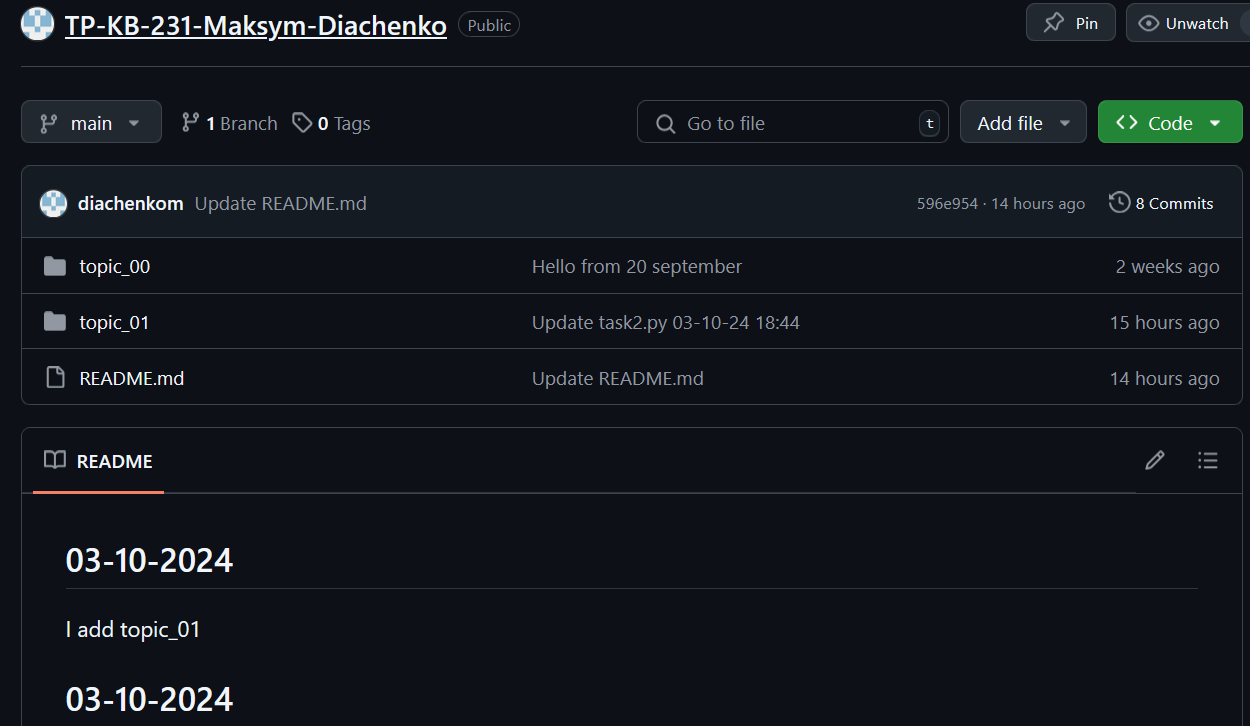
d = discr(a, b, c)

print("Discriminant is: " + str(d))

Посилання на github:

<https://github.com/diachenkom/TP-KB-231-Maksym-Diachenko>

Знімок екрану з посилання на github:



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

Звіт до Теми №2

Умовні переходи

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

1) Написати функцію пошуку коренів квадратного рівняння використовуючи функцію розрахунку дискримінанту з попередньої теми та умовні переходи.

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

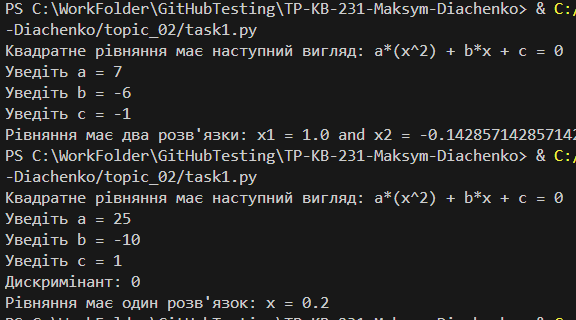
3) Написати програму калькулятор використовуючи match конструкцію. Кожна операція має бути виконана в окремій функції.

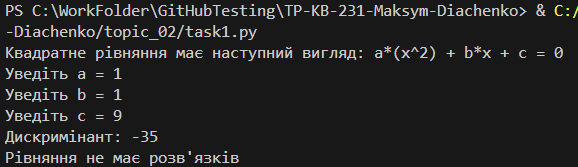
**Функція пошуку коренів квадратного рівняння**

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

Хід виконання завдання:

У цьому завданні я, використовуючи функцію розрахунку дискримінанту з попередньої теми, написав функцію пошуку коренів квадратного рівняння з використанням умовних переходів.





Текст програми:

import math

print("Квадратне рівняння має наступний вигляд: a\*(x^2) + b\*x + c = 0")

a = int(input("Уведіть a = "))

b = int(input("Уведіть b = "))

c = int(input("Уведіть c = "))

def discr(a, b, c):

    return b\*\*2 - 4\*a\*c

def equation\_solution(a, b, c):

    d = discr(a, b, c)

    print("Дискримінант: " + str(d))

    if d > 0:

        x1 = (-(b) + math.sqrt(d)) / (2\*a)

        x2 = (-(b) - math.sqrt(d)) / (2\*a)

        return f"Рівняння має два розв'язки: x1 = {x1} and x2 = {x2}"

    elif d == 0:

        x1 = (-b) / (2\*a)

        return f"Рівняння має один розв'язок: x = {x1}"

    else:

        return "Рівняння не має розв'язків"

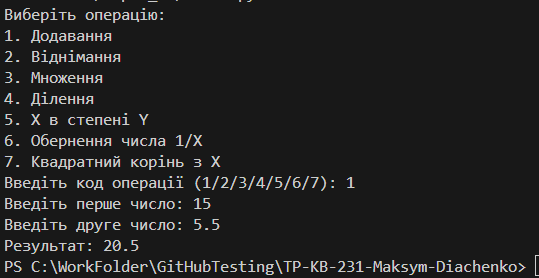
print (equation\_solution(a, b, c))

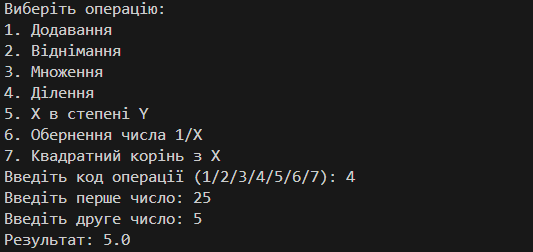
**Программа калькулятор з if else конструкцію**

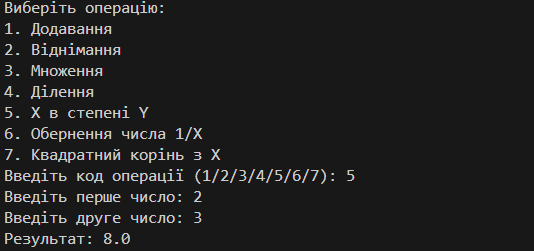
Необхідно написати програму калькулятор використовуючи if else конструкцію.

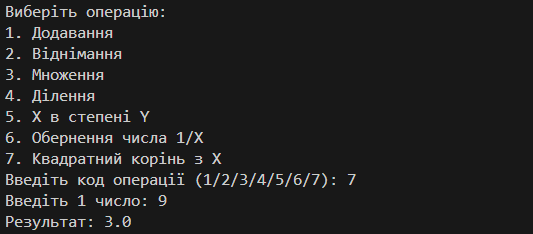
Хід виконання завдання:

У цьому завданні я написав програму калькулятор, що використовує if else конструкції, оформивши кожну операцію в окремій функції.









Текст програми:

# Додавання

def add(a, b):

    return a + b

# Віднімання

def subtract(a, b):

    return a - b

# Множення

def multiply(a, b):

    return a \* b

# Ділення

def divide(a, b):

    if b != 0:

        return a / b

    else:

        return "Помилка: Ділити на нуль не можна!"

# X в степені Y

def step(a,b):

    return a\*\*b

# Основна функція калькулятора

def calculator():

    print("Виберіть операцію:")

    print("1. Додавання")

    print("2. Віднімання")

    print("3. Множення")

    print("4. Ділення")

    print("5. X в степені Y")

    print("6. Обернення числа 1/Х")

    print("7. Квадратний корінь з Х")

    cod = input("Введіть код операції (1/2/3/4/5/6/7): ")

    if (int(cod)<1 or int(cod)>7):

        print("Неправильний ввiд Kоду операції")

        return

    elif (int(cod)>=1 and int(cod)<=5):

        num1 = float(input("Введіть перше число: "))

        num2 = float(input("Введіть друге число: "))

    elif (int(cod)>=6 and int(cod)<=7):

        num1 = float(input("Введіть 1 число: "))

    # Виконання арифметичної операції

    if cod == '1':

        print(f"Результат: {add(num1, num2)}")

    elif cod == '2':

        print(f"Результат: {subtract(num1, num2)}")

    elif cod == '3':

        print(f"Результат: {multiply(num1, num2)}")

    elif cod == '4':

        print(f"Результат: {divide(num1, num2)}")

    elif cod == '5':

        print(f"Результат: {step(num1, num2)}")

    elif cod == '6':

        print(f"Результат: {1/num1}")

    elif cod == '7':

        print(f"Результат: {num1\*\*0.5}")

    else:

        print("Неправильний код операції.")

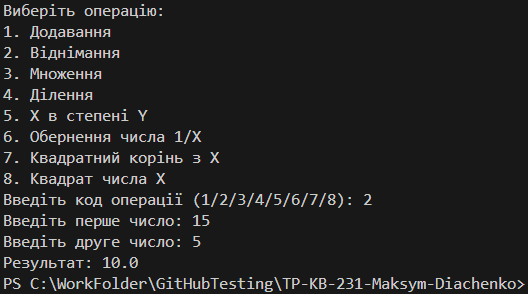
calculator()

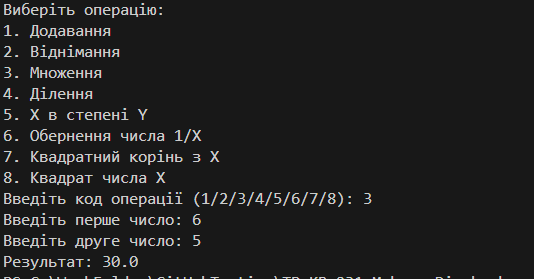
**Програма калькулятор з match конструкцією**

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

Хід виконання завдання:

У цьому завданні я написав програму калькулятор, аналогічно до попередньої. Але в цьому завданні вже з використанням match конструкції, оформивши кожну операцію в окремій функції.





Текст програми:

# Додавання

def add(a, b):

    return a + b

# Віднімання

def subtract(a, b):

    return a - b

# Множення

def multiply(a, b):

    return a \* b

# Ділення

def divide(a, b):

    if b != 0:

        return a / b

    else:

        return "Помилка: Ділити на нуль не можна!"

# X в степені Y

def step(a,b):

    return a\*\*b

def square(a):

    return a\*\*2

# Основна функція калькулятора

def calculator():

    print("Виберіть операцію:")

    print("1. Додавання")

    print("2. Віднімання")

    print("3. Множення")

    print("4. Ділення")

    print("5. X в степені Y")

    print("6. Обернення числа 1/Х")

    print("7. Квадратний корінь з Х")

    print("8. Квадрат числа Х")

    cod = input("Введіть код операції (1/2/3/4/5/6/7/8): ")

    if (int(cod)<1 or int(cod)>8):

        print("Неправильний ввiд Kоду операції")

        return

    elif (int(cod)>=1 and int(cod)<=5):

        num1 = float(input("Введіть перше число: "))

        num2 = float(input("Введіть друге число: "))

    elif (int(cod)>=6 and int(cod)<=8):

        num1 = float(input("Введіть 1 число: "))

    # Виконання арифметичної операції

    match cod:

        case '1':

            print(f"Результат: {add(num1, num2)}")

        case '2':

            print(f"Результат: {subtract(num1, num2)}")

        case '3':

            print(f"Результат: {multiply(num1, num2)}")

        case '4':

            print(f"Результат: {divide(num1, num2)}")

        case '5':

            print(f"Результат: {step(num1, num2)}")

        case '6':

            print(f"Результат: {1/num1}")

        case '7':

            print(f"Результат: {num1\*\*0.5}")

        case '8':

            print(f"Результат: {square(num1)}")

        case \_:

            print("Неправильний код операції.")

calculator()

Посилання на github:

<https://github.com/diachenkom/TP-KB-231-Maksym-Diachenko>

Знімок екрану з посилання на github: