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

f

Звіт до Теми №2

Умовний перехід

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

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

**Текст програми:**

|  |
| --- |
| from math import \*  def discriminant(a, b, c): # Пошук дискримінанта      return b\*\*2 - 4 \* a \* c  def quadratic(a, b, c):      D = discriminant(a, b, c)        if D > 0: # Якщо дискримінант більше 0, то має два кореня          x1 = (-b + sqrt(D)) / (2\*a)          x2 = (-b - sqrt(D)) / (2\*a)          return x1, x2      elif D == 0: # Якщо дискримінант == 0, то має один корень          x = -b / (2 \* a)          return x, x      else: # Якщо дискримінант < 0, то немає коренів          return "Коренів немає"  print(quadratic(-2, 5, 6)) |

**Посилання на github:**

<https://github.com/Alexxxxo/TP-KB-231-Oleksandr-Kozachok/blob/main/topic_02/task1.py>

**Знімок екрану з посилання на github:**

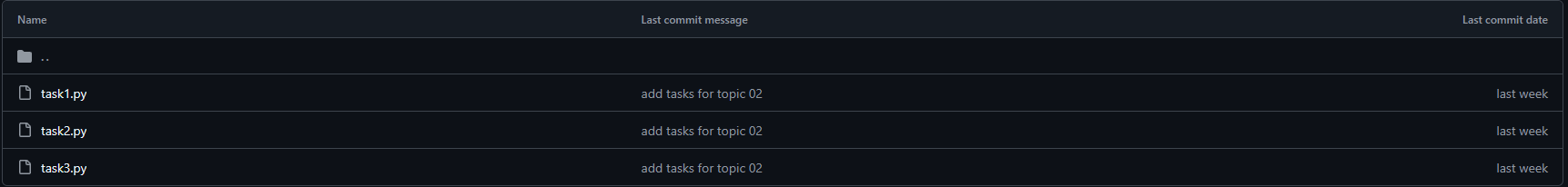


Рисунок 1.1 – Знімок екрану з посилання на GitHub.

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

**Текст програми:**

|  |
| --- |
| def add(a, b):      return a + b  def subtract(a, b):      return a - b  def multiply(a, b):      return a \* b  def divide(a, b):      if b != 0:          return a / b      else:          return "На нуль ділити не можна"  def checkOper(operation):      return operation in "+-\*/"  def calc():      while True:          expression = input("Введіть 'exit' для виходу з программи або будь-яке число: ")          if expression.lower() == "exit":              print("бай-бай")              break            num1 = float(expression)          operation = input("Введіть операцію (+, -, \*, /): ")            if not checkOper(operation):              print("Неправильна операція")              continue            num2 = float(input("Введіть друге число: "))            if operation == "+":              result = add(num1, num2)          elif operation == "-":              result = subtract(num1, num2)          elif operation == "\*":              result = multiply(num1, num2)          elif operation == "/":              result = divide(num1, num2)          print("Result =", result)  calc() |

**Посилання на github:**

<https://github.com/Alexxxxo/TP-KB-231-Oleksandr-Kozachok/blob/main/topic_02/task2.py>

**Знімок екрану з посилання на github:**

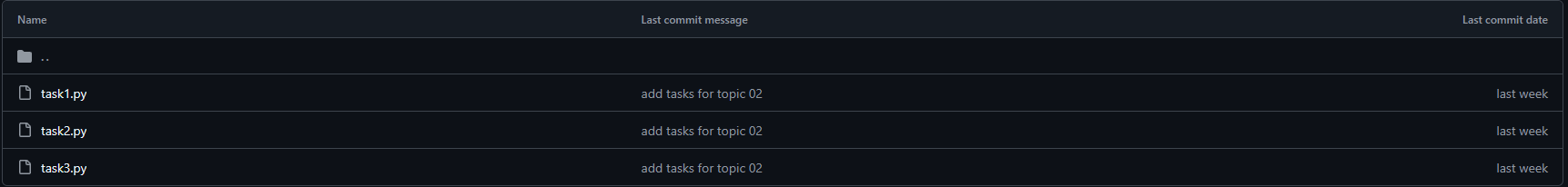


Рисунок 1.2 - Знімок екрану з посилання на GitHub.

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

**Текст програми:**

|  |
| --- |
| def add(a, b):      return a + b  def subtract(a, b):      return a - b  def multiply(a, b):      return a \* b  def divide(a, b):      if b != 0:          return a / b      else:          return "На нуль ділити не можна"  def checkOper(operation):      return operation in "+-\*/"  def calc():      while True:          expression = input("Введіть 'exit' для виходу з программи або будь-яке число: ")          if expression.lower() == "exit":              print("бай-бай")              break            num1 = float(expression)          operation = input("Введіть операцію (+, -, \*, /): ")            if not checkOper(operation):              print("Неправильна операція")              continue            num2 = float(input("Введіть друге число: "))          match operation:              case '+':                  result = add(num1, num2)              case '-':                  result = subtract(num1, num2)              case '\*':                  result = multiply(num1, num2)              case '/':                  result = divide(num1, num2)          print("Result =", result)  calc() |

**Посилання на github:**

<https://github.com/Alexxxxo/TP-KB-231-Oleksandr-Kozachok/blob/main/topic_02/task3.py>

**Знімок екрану з посилання на github:**

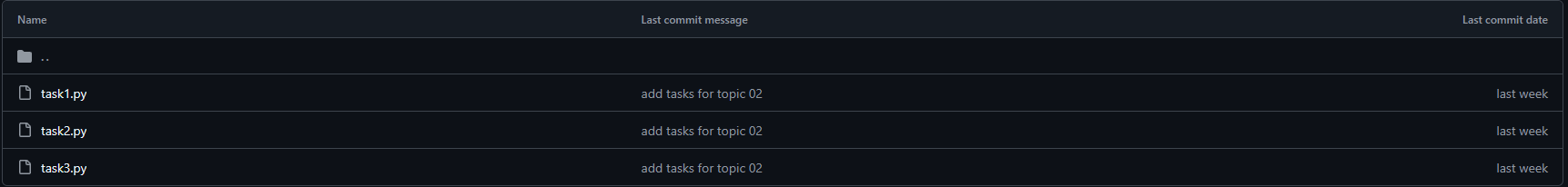


Рисунок 1.3 - Знімок екрану з посилання на GitHub.