Міністерство освіти і науки України

Національний університет «Чернігівська політехніка»

Навчально-науковий інститут електронних та інформаційних технологій

Кафедра кібербезпеки та математичного моделювання

**Звіт про виконання практичних завдань**

**до лекцій з курсу:**

**«Технології програмування на мові Python»**

виконала

студентка групи КБ-232

Ребенок Є. О.

перевірив

Дюба І.М\_\_\_\_\_\_\_\_\_

2024

**Зміст**

[**Звіт до Теми №1** 3](#_Toc178842085)

[**Звіт до Теми №2** 7](#_Toc178842086)

# **Звіт до Теми №1**

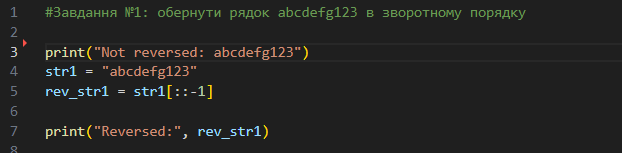
Функції та змінні

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

**Перетворення рядка**

Необхідно рядок, що має вигляд "abcdefg123" перетворити наступним чином "321gfedcba", вважаючи сталою довжину рядку в 10 символів.

Хід виконання завдання:



Вивід:



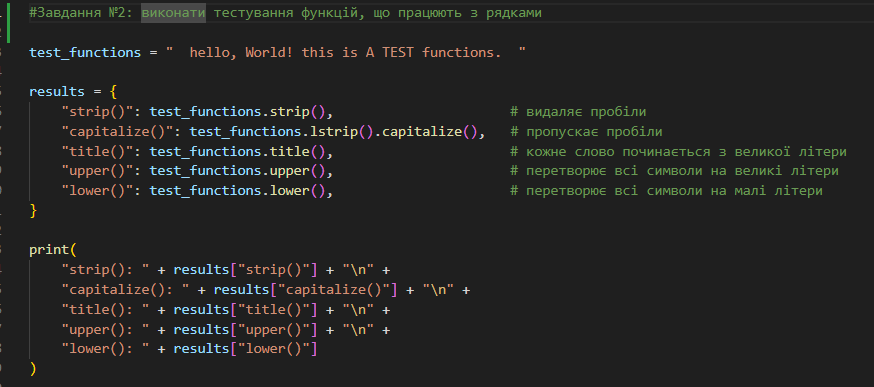
Текст програми:

|  |
| --- |
| print("Not reversed: abcdefg123")  str1 = "abcdefg123"  rev\_str1 = str1[::-1]  print("Reversed:", rev\_str1) |

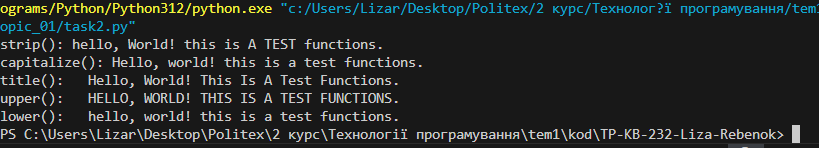
**Робота з рядками**

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

Хід виконання завдання:



Вивід:



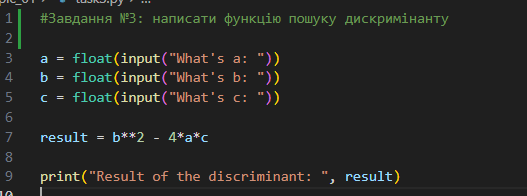
Текст програми:

|  |
| --- |
| test\_functions = " hello, World! this is A TEST functions. "  results = {  "strip()": test\_functions.strip(), # видаляє пробіли  "capitalize()": test\_functions.lstrip().capitalize(),# пропускає пробіли  "title()": test\_functions.title(), # кожне слово починається з великої літери  "upper()": test\_functions.upper(), # перетворює всі символи на великі літери  "lower()": test\_functions.lower(), # перетворює всі символи на малі літери  }  print(  "strip(): " + results["strip()"] + "\n" +  "capitalize(): " + results["capitalize()"] + "\n" +  "title(): " + results["title()"] + "\n" +  "upper(): " + results["upper()"] + "\n" +  "lower(): " + results["lower()"]  ) |

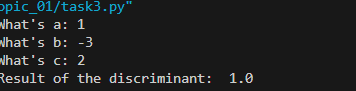
**Пошук дискримінанта квадратного рівняння**

Написати функцію пошуку дискримінанту квадратного рівняння.

Хід виконання завдання:



Вивід:



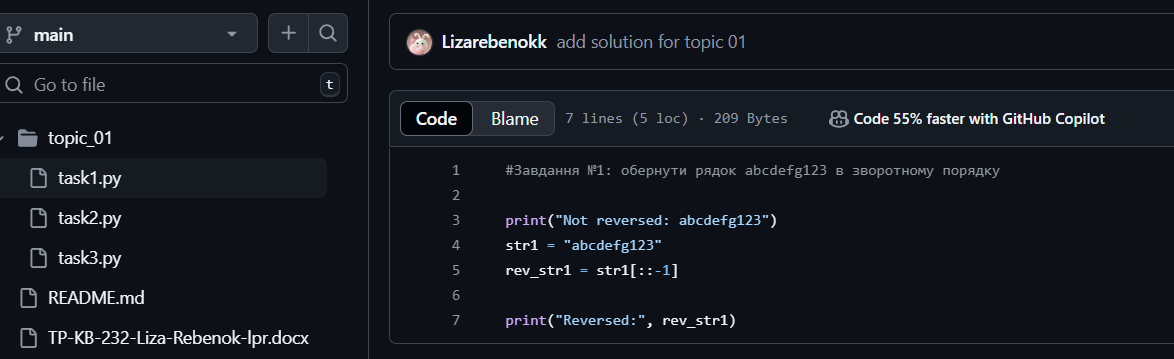
Текст програми:

|  |
| --- |
| a = float(input("What's a: "))  b = float(input("What's b: "))  c = float(input("What's c: "))  result = b\*\*2 - 4\*a\*c  print("Result of the discriminant: ", result) |

Посилання на github:

<https://github.com/Lizarebenokk>

Знімок екрану з посилання на github:



# **Звіт до Теми №2**

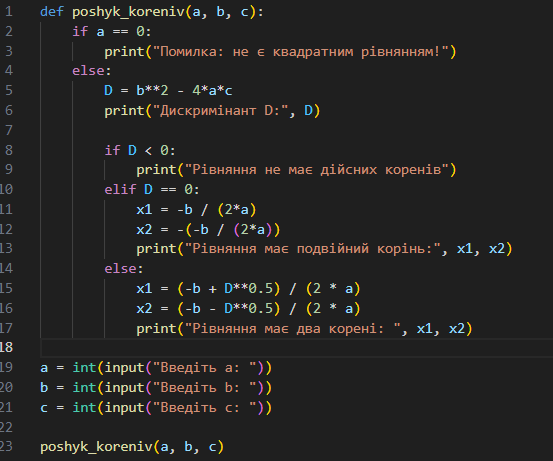
Умовний перехід

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

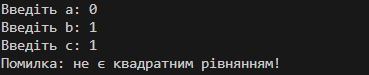
**Написання функцій**

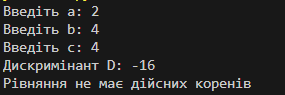
Написати функцію пошуку коренів квадратного рівняння використовуючи функцію розрахунку дискримінанту з попередньої теми та умовні переходи.

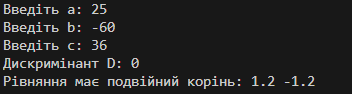
Хід виконання завдання:

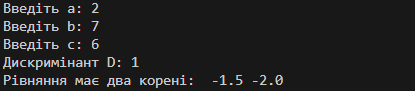


Вивід:









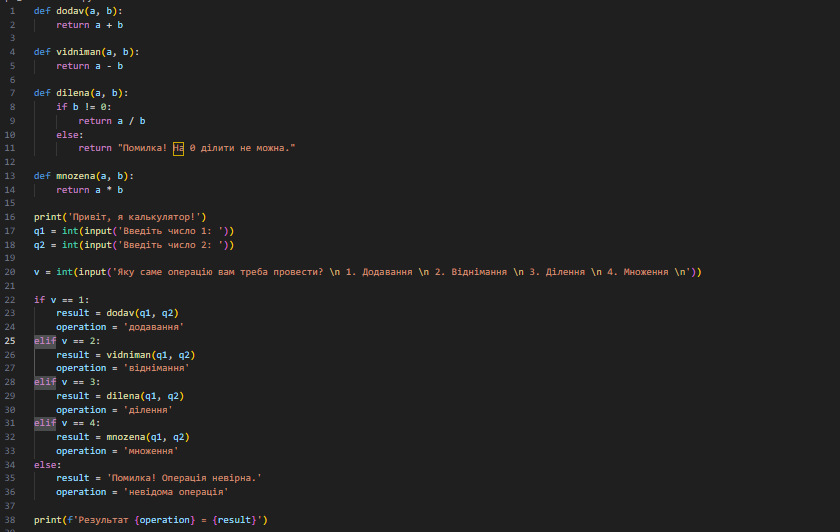
Текст програми:

|  |
| --- |
| def poshyk\_koreniv(a, b, c):  if a == 0:  print("Помилка: не є квадратним рівнянням!")  else:  D = b\*\*2 - 4\*a\*c  print("Дискримінант D:", D)  if D < 0:  print("Рівняння не має дійсних коренів")  elif D == 0:  x1 = -b / (2\*a)  x2 = -(-b / (2\*a))  print("Рівняння має подвійний корінь:", x1, x2)  else:  x1 = (-b + D\*\*0.5) / (2 \* a)  x2 = (-b - D\*\*0.5) / (2 \* a)  print("Рівняння має два корені: ", x1, x2)  a = int(input("Введіть a: "))  b = int(input("Введіть b: "))  c = int(input("Введіть c: "))    poshyk\_koreniv(a, b, c) |

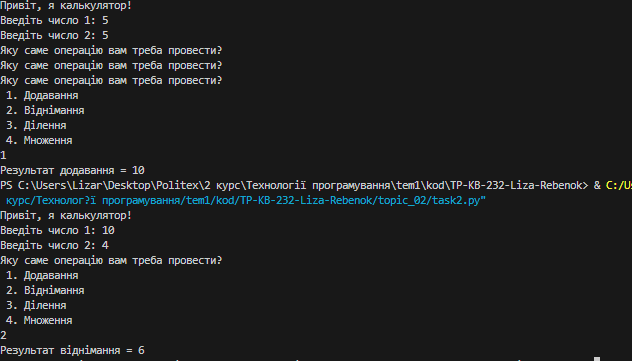
**Написання програми калькулятор**

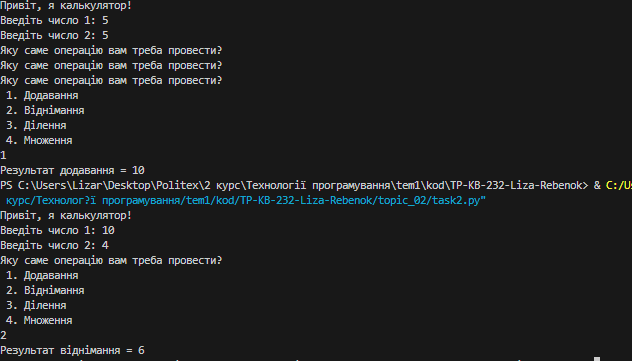
Написати програму калькулятор використовуючи if else конструкцію. Кожна операція має бути виконана в окремій функції.

Хід виконання завдання:



Вивід:



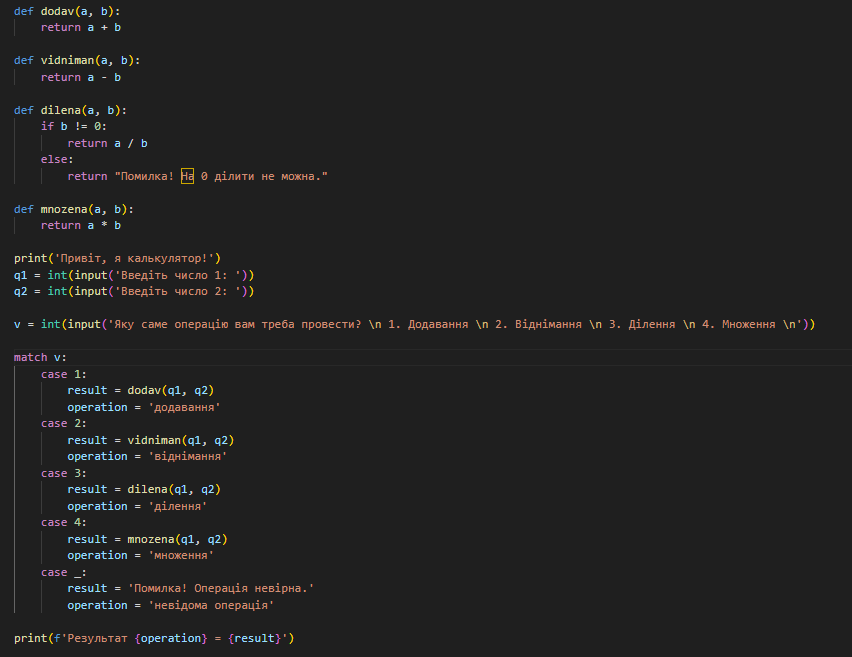


Текст програми:

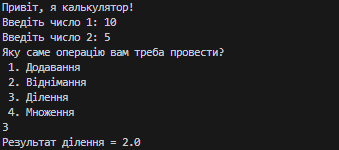
|  |
| --- |
| def dodav(a, b):  return a + b  def vidniman(a, b):  return a - b  def dilena(a, b):  if b != 0:  return a / b  else:  return "Помилка! На 0 ділити не можна."  def mnozena(a, b):  return a \* b  print('Привіт, я калькулятор!')  q1 = int(input('Введіть число 1: '))  q2 = int(input('Введіть число 2: '))  v = int(input('Яку саме операцію вам треба провести? \n 1. Додавання \n 2. Віднімання \n 3. Ділення \n 4. Множення \n'))  if v == 1:  result = dodav(q1, q2)  operation = 'додавання'  elif v == 2:  result = vidniman(q1, q2)  operation = 'віднімання'  elif v == 3:  result = dilena(q1, q2)  operation = 'ділення'  elif v == 4:  result = mnozena(q1, q2)  operation = 'множення'  else:  result = 'Помилка! Операція невірна.'  operation = 'невідома операція'  print(f'Результат {operation} = {result}') |

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

Хід виконання завдання:



Вивід:



Текст програми:

|  |
| --- |
| def dodav(a, b):  return a + b  def vidniman(a, b):  return a - b  def dilena(a, b):  if b != 0:  return a / b  else:  return "Помилка! На 0 ділити не можна."  def mnozena(a, b):  return a \* b  print('Привіт, я калькулятор!')  q1 = int(input('Введіть число 1: '))  q2 = int(input('Введіть число 2: '))  v = int(input('Яку саме операцію вам треба провести? \n 1. Додавання \n 2. Віднімання \n 3. Ділення \n 4. Множення \n'))  match v:  case 1:  result = dodav(q1, q2)  operation = 'додавання'  case 2:  result = vidniman(q1, q2)  operation = 'віднімання'  case 3:  result = dilena(q1, q2)  operation = 'ділення'  case 4:  result = mnozena(q1, q2)  operation = 'множення'  case \_:  result = 'Помилка! Операція невірна.'  operation = 'невідома операція'  print(f'Результат {operation} = {result}') |

Посилання на github:

<https://github.com/Lizarebenokk>

Знімок екрану з посилання на github:

