мІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

нАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «лЬВІВСЬКА ПОЛІТЕХНІКА»

ІНСТИТУТ КОМП’ЮТЕРНИХ НАУК ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

|  |
| --- |
| 2023 |



Звіт

Про виконання лабораторної роботи №6

З предмету «Спеціалізовані мови програмування»

Виконав

ст.гр ІТ-21

Капанайко А.Т

Прийняв:

доц.каф

Щербак С.С

**Мета роботи:** Cтворення юніт-тестів для додатка-калькулятора на основі класів

**План роботи:**

**Завдання 1:** Тестування Додавання

Напишіть юніт-тест, щоб перевірити, що операція додавання в вашому додатку-калькуляторі працює правильно. Надайте тестові випадки як для позитивних, так і для негативних чисел.

**Завдання 2:** Тестування Віднімання

Створіть юніт-тести для переконання, що операція віднімання працює правильно. Тестуйте різні сценарії, включаючи випадки з від'ємними результатами.

**Завдання 3:** Тестування Множення

Напишіть юніт-тести, щоб перевірити правильність операції множення в вашому калькуляторі. Включіть випадки з нулем, позитивними та від'ємними числами.

**Завдання 4:** Тестування Ділення

Розробіть юніт-тести для підтвердження точності операції ділення. Тести повинні охоплювати ситуації, пов'язані з діленням на нуль та різними числовими значеннями.

**Завдання 5:** Тестування Обробки Помилок

Створіть юніт-тести, щоб перевірити, як ваш додаток-калькулятор обробляє помилки. Включіть тести для ділення на нуль та інших потенційних сценаріїв помилок. Переконайтеся, що додаток відображає відповідні повідомлення про помилки.

**Код програми:**

import math

import unittest

import sys

from importlib.machinery import SourceFileLoader

sys.path.append(r"C:\Users\super\Desktop\3kurs\1sem\smp\lab1-on-python\lab2")

lab2\_module = SourceFileLoader(

    "calculator", r"C:\Users\super\Desktop\3kurs\1sem\smp\lab1-on-python\lab2\lab2.py"

).load\_module()

class CalculatorTests(unittest.TestCase):

    def setUp(self):

        self.calculator = lab2\_module.Calculator()

    def test\_addition(self):

        self.calculator.value1 = 5

        self.calculator.value2 = 3

        self.calculator.operator = "+"

        self.calculator.calculation()

        self.assertEqual(self.calculator.result, 8)

    def test\_subtraction(self):

        self.calculator.value1 = 5

        self.calculator.value2 = 3

        self.calculator.operator = "-"

        self.calculator.calculation()

        self.assertEqual(self.calculator.result, 2)

    def test\_multiplication(self):

        self.calculator.value1 = 5

        self.calculator.value2 = 3

        self.calculator.operator = "\*"

        self.calculator.calculation()

        self.assertEqual(self.calculator.result, 15)

    def test\_division(self):

        self.calculator.value1 = 6

        self.calculator.value2 = 2

        self.calculator.operator = "/"

        self.calculator.calculation()

        self.assertEqual(self.calculator.result, 3)

    def test\_division\_by\_zero(self):

        self.calculator.value1 = 6

        self.calculator.value2 = 0

        self.calculator.operator = "/"

        with self.assertRaises(ZeroDivisionError):

            self.calculator.calculation()

if \_\_name\_\_ == "\_\_main\_\_":

    unittest.main()

**Посилання на GitHub репозиторій:** [**https://github.com/Senichkaa/lab1-on-python**](https://github.com/Senichkaa/lab1-on-python)

**Висновок:** Виконавши ці завдання, у вас буде набір юніт-тестів, які перевіряють правильність основних арифметичних операцій у вашому додатку-калькуляторі. Ці тести допоможуть виявити та виправити будь-які проблеми або помилки, які можуть виникнути під час розробки чи обслуговування вашого додатку, забезпечуючи його надійність і точність