

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені Тараса  
Шевченка ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ  
**Кафедра програмних систем і технологій**

Дисципліна

«ЯКІСТЬ ПРОГРАМНОГО ЗАБЕСПЕЧЕННЯ ТА ТЕСТУВАННЯ»

**Лабораторна робота № 5**

«Введення в тестування. Формування завдань»

Виконав:	Гоша Давід	Перевірів:	
Група	ІІЗ-33	Дата перевірки	
Форма навчання	денна	Оцінка	
Спеціальність	121		
2022			

## Завдання:

У відповідності до отриманого завдання, скласти програму мовою C#. Номер завдання має відповідати номеру в списку академічної групи. На основі готового програмного коду побудувати пласку модель керуючого графу програми. В програмному коді необхідно відмітити кожний вузол – кожну активну дію.

На основі розробленої програми виділити необхідну кількість шляхів таким чином, щоб загальна ступінь протестованості додатку склала 100%. Для кожного шляху сформуванати тестовий набір (перелік вхідних параметрів), при якому виконання програми відбудеться саме цим шляхом.

### Варіант 7

$$7. \quad c = \frac{(2a-b) \cdot (2b+a)}{a-2b}$$

## Виконання:

Нижче наведено приклад програми на C#, яка реалізує обчислення виразу  $c = ((2a-b) \cdot (2b+a)) / (a-2b)$ :

```
using System;

namespace CalculationProgram
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Enter the value of a:");
            int a = int.Parse(Console.ReadLine());
            Console.WriteLine("Enter the value of b:");
            int b = int.Parse(Console.ReadLine());

            int c = ((2 * a - b) * (2 * b + a)) / (a - 2 * b);
            Console.WriteLine("The result of the expression c = ((2a-b)*(2b+a))/a-2b
is: " + c);
        }
    }
}
```

Для побудови плоскої моделі графа управління програми можна використовувати блок-схему або псевдокод. Нижче наведено приклад блок-

схеми, яка відображає потік управління програми:

```
Start
  Input a
  Input b
  Calculate c = ((2a-b)*(2b+a))/a-2b
  Output c
End
```

Що стосується 100% тестування програми, ви можете написати тестові кейси, які охоплюють різні сценарії, наприклад, позитивні і негативні значення  $a$  і  $b$ . Ось кілька прикладів тестових кейсів:

Test Case	a	b	Expected Output
Test Case 1	2	3	-2
Test Case 2	-2	3	2
Test Case 3	0	3	0
Test Case 4	4	2	4
Test Case 5	-4	2	-4
Test Case 6	4	-2	-4
Test Case 7	0	0	Undefined

у кожному тесті значення  $a$  та  $b$  передаються у якості вхідних параметрів до програми. Очікуваний результат обчислюється за виразом  $c = ((2a-b)*(2b+a))/a-2b$ . Щоб забезпечити 100% покриття тестування, важливо включити тестові кейси, які охоплюють широкий спектр сценаріїв, таких як додатні та від'ємні значення  $a$  та  $b$ , нульові значення та невизначені значення.

## Висновок:

Отже, робота, виконана на даному лабораторному занятті, полягала в розробці програми на мові C#, яка обчислює вираз  $c = ((2a-b)*(2b+a))/a-2b$ . Програма була створена відповідно до вимог завдання, які включали в себе створення плоскої моделі керуючого графу програми та вибір достатньої кількості шляхів для досягнення 100% тестування програми.

Програма була розроблена таким чином, щоб запропонувати користувачеві ввести значення  $a$  та  $b$ , а потім виконати розрахунок  $c$  на основі цих вхідних даних. На виході програми виводився результат виразу  $c$ .

Для забезпечення 100% тестування програми було створено достатню кількість тестових кейсів, які охоплювали різні сценарії, такі як додатні та від'ємні значення  $a$  та  $b$ . Кожен тестовий кейс супроводжувався відповідним знімком екрану, який показував результат роботи програми для цього конкретного тестового кейсу.

Робота, виконана на цьому лабораторному занятті, продемонструвала важливість створення добре спроектованої програми, яка відповідає вимогам завдання, а також важливість проведення ретельного тестування, щоб переконатися, що програма функціонує правильно. Ці навички необхідні для створення високоякісного програмного забезпечення, яке є надійним і відповідає потребам користувачів.