

Гоша Давід Олександрович

Лабораторна робота № 7

Завдання:

- Ознайомитися з теоретичними відомостями по стратегіях тестування.
- Відповідно до варіанту задачі, підготувати тести по методиках стратегії "чорного ящика".
- Пропоновані тести звести в таблицю. Номер тесту Призначення тесту Значення початкових даних Очікуваний результат Реакція програми Висновок
- Розробити програму.
- Виконати тестування. Занести в таблицю результати.
- Зробити висновок про роль тестування з використанням стратегій "чорного ящика" і можливості його вживання. Сформулювати його достоїнства і недоліки.

Задача 1

Код:

Розробити програму рішення квадратного рівняння, де a, b, c - будь-які дійсні числа.

```
import cmath

def quadratic_solver(a, b, c):
    if a == 0:
        raise ValueError("The coefficient 'a' must not be zero.")

    # Calculate the discriminant
    discriminant = cmath.sqrt(b * b - 4 * a * c)

    # Calculate the two solutions
    x1 = (-b + discriminant) / (2 * a)
    x2 = (-b - discriminant) / (2 * a)

    return (x1, x2)

if __name__ == "__main__":
    a = float(input("Enter the coefficient a: "))
    b = float(input("Enter the coefficient b: "))
    c = float(input("Enter the coefficient c: "))

    try:
        x1, x2 = quadratic_solver(a, b, c)
        print(f"The solutions are: x1 = {x1}, x2 = {x2}")
    except ValueError as e:
        print(e)
```

Таблиця тестування:

Номер тесту	Мета тесту	Початкові дані (a, b, c)	Очікуваний результат	Реакція програми
1	Дійсні корені (розбиття на частини)	(1, -3, 2)	x1 = 1, x2 = 2	The solutions are: x1 = (2+0j), x2 = (1+0j)
2	Комплексні корені	(1, 2, 2)	x1 = -1 + i, x2 = -1 - i	The solutions are: x1 = (-1+1j), x2 = (-1-1j)

	(розбиття на частини)			
3	Єдиний дійсний корінь (розбиття на частини)	(1, -4, 4)	$x_1 = x_2 = 2$	The solutions are: $x_1 = (2+0j)$, $x_2 = (2+0j)$
4	Близькі до нуля (Аналіз граничних значень)	(0.0001, 0.0001, -0.1)	Дійсні корені (точні значення залежать від реалізації)	The solutions are: $x_1 = (31.12672920173694+0j)$, $x_2 = (-32.12672920173694+0j)$
5	Максимальна межа (Аналіз граничних значень)	(MAX, MAX, MAX)	Комплексні корені (точні значення залежать від реалізації)	The solutions are: $x_1 = (-0.5+0.8660254037844387j)$, $x_2 = (-0.5-0.8660254037844387j)$
6	$a = 0$ (Причинно-наслідковий аналіз)	(0, 2, -1)	Дискримінант = 0: (1, -6, 9)	The coefficient 'a' must not be zero.

Висновок:

Отже, тестування за допомогою "чорного ящика" є важливою частиною процесу розробки програмного забезпечення, оскільки воно допомагає виявити потенційні помилки і гарантує, що програма відповідає бажаній функціональності. Використовуючи методи тестування "чорного ящика", такі як розбиття на частини, аналіз граничних значень та причинно-наслідковий аналіз, ми можемо ефективно тестувати програму, не беручи до уваги її внутрішню реалізацію.

Для програми розв'язування квадратних рівнянь ми підготували різні тестові кейси для різних типів коренів (дійсні, комплексні, єдиний дійсний корінь) та граничних випадків (коефіцієнти близькі до нуля та максимальна межа). Ці тести допомогли нам переконатися, що програма працює як очікувалося і може обробляти різні типи вхідних даних.

Переваги тестування за допомогою чорної скриньки:

- 1. Тестувальникам не потрібні знання внутрішньої реалізації програми.
- 2. Вони зосереджуються на функціональності та гарантують, що програма відповідає вимогам користувача.
- 3. Тести є об'єктивними, оскільки вони розробляються на основі вимог і специфікацій, а не коду.

Недоліки тестування чорного ящика:

- 1. Воно може не охоплювати всі можливі сценарії, оскільки основна увага приділяється функціональності, а не внутрішній логіці.
- 2. Деякі тести можуть бути надлишковими, якщо одна і та ж частина коду тестується з різними вхідними даними.
- 3. Він може бути менш ефективним для більш складних програм, оскільки не враховує внутрішню структуру і логіку програми.

Загалом, тестування чорного ящика є цінною технікою, яка доповнює інші методи тестування, такі як тестування білого ящика, щоб забезпечити комплексну оцінку якості та функціональності програми.

