**Отчет о тестировании проекта "Lab2\_KosinovaAi"**

a. Количество реализованных тестов, статистика прохождения тестов для каждого из проектов:

Общее количество запущенных тестов: 16

Пройдено успешно: 11 тестов

Не пройдено: 5 тестов

Пропущено: 0 тестов

Детализация прохождения тестов:

TriangleTests.TestValidFloatingPointSides [FAIL]

Ожидаемый результат: "Разносторонний"

Фактический результат: (не указан)

TriangleTests.TestValidExcessivelyLargeSides [FAIL]

Ожидаемый результат: "Разносторонний"

Фактический результат: "Не треугольник"

TriangleTests.TestValidTriangle [FAIL]

Ожидаемый результат: 3

Фактический результат: 0

TriangleTests.TestValidSmallSides [FAIL]

Ожидаемый результат: "Разносторонний"

Фактический результат: (не указан)

TriangleTests.TestValidLargeSides [FAIL]

Ожидаемый результат: "Разносторонний"

Фактический результат: "Не треугольник"

b. Количество не проходящих тестов для проекта: 5 тестов.

c. Локализации кода, которые, возможно, приводят к непрохождению тестов:

private string DetermineTriangleType(float a, float b, float c)

{

// Определение типа треугольника

if (a == b && b == c)

return "Равносторонний";

else if (a == b || a == c || b == c)

return "Равнобедренный";

else

return "Разносторонний";

}

Данный участок кода может некорректно отображать результаты ввиду некоторых причин:  
1. Ожидаемые значения в тестах: если ожидается, что метод вернет конкретное значение (например, "Разносторонний"), но фактический результат отличается от ожидаемого, то это может привести к ошибке в тестах

2. Округление: Вещественные числа (тип float) могут быть представлены с погрешностью. Это может вызвать проблемы при сравнении их на равенство (например, a == b). Вместо точного сравнения стоит использовать погрешность, например, сравнивать их с небольшой допустимой погрешностью.

3. Регистр символов: если ожидаемые значения имеют определенный регистр символов (например, "Разносторонний" с большой буквы), а метод возвращает значения с другим регистром

Неучтенные краевые случаи: Метод должен учитывать все возможные варианты ввода и обработки сторон треугольника. Это включает в себя не только обычные случаи, но и краевые случаи, такие как нулевые стороны или очень маленькие/большие значения.