Лабораторная работа №5

“Исследование способов модульного тестирования программного

обеспечения в среде NUnit”

1 ЦЕЛЬ РАБОТЫ

Исследовать эффективность использования методологии TDD при разработке программного обеспечения. Получить практические навыки использования фреймворка NUnit для модульного тестирования программного обеспечения.

2 ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Реализовать на языке C# один из классов, спроектированных в лабораторной работе № 1. Методы класса при этом не реализовывать.
2. Разработать для созданного класса набор модульных тестов, включающий тесты для каждого метода.
3. Запустить набор тестов, проанализировать и сохранить результаты.
4. Поочередно реализовать методы класса, выполняя тестирование при каждом изменении программного кода.
5. После того, как весь набор тестов будет выполняться успешно, реализацию классов можно считать завершенной.

3 ТЕКСТ ПРОГРАММЫ

Aster.cs:

using System;

namespace AsterLib

{

public class Aster

{

int k = 0;

private string str;

public Aster(string ss)

{

str = ss;

}

public string Replace(string str)

{

for (int i = 0; i < str.Length; i++)

{

if (i == str.Length - 1)

{

Console.WriteLine("Знак '\*' находится в конце, замена невозможна");

break;

}

else

if (str[i] == '\*')

{

k++;

char[] charStr = str.ToCharArray();

charStr[i + 1] = '-';

str = new string(charStr);

}

}

return str;

}

public string SuccessCheck()

{

Replace(str);

if (k > 0)

{

str = "Success: " + str;

Console.WriteLine("Итоговая строка: " + str);

}

else Console.WriteLine("Знак '\*' не найден");

return str;

}

}

}

AsterTest.cs:

using NUnit.Framework;

namespace AsterLib

{

[TestFixture]

public class AsterTest

{

string[] strings = {

"Первая строка",

"\*Вторая строка",

"Третья строка\*",

"Четвертая строк\*а",

"Пят\*ая ст\*рока",

"Шест\*\*ая строка",

"\*\*\*\*\*\*\*\*"

};

string[] compare = {

"Первая строка",

"\*-торая строка",

"Третья строка\*",

"Четвертая строк\*-",

"Пят\*-я ст\*-ока",

"Шест\*-ая строка",

"\*-\*-\*-\*-"

};

[Test]

public void Test1()

{

for (int i = 0; i < strings.Length; i++)

{

Aster strs = new Aster(null);

string result = strs.Replace(strings[i]);

Assert.AreEqual(result, compare[i]);

}

}

[Test]

public void Test2()

{

for (int i = 0; i < strings.Length; i++)

{

Aster strs = new Aster(strings[i]);

string result = strs.SuccessCheck();

Assert.AreEqual("Success: " + result, compare[i]);

}

}

}

}

4 РЕЗУЛЬТАТЫ

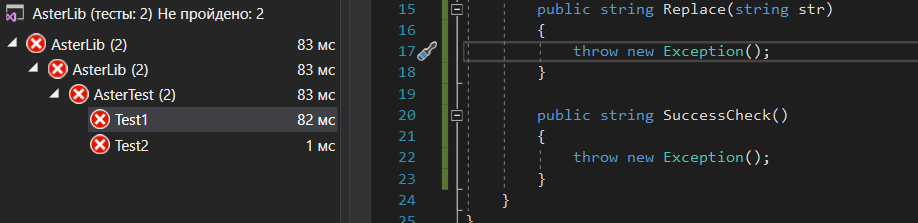


Рисунок 1 – Результаты тестирования без реализованных методов

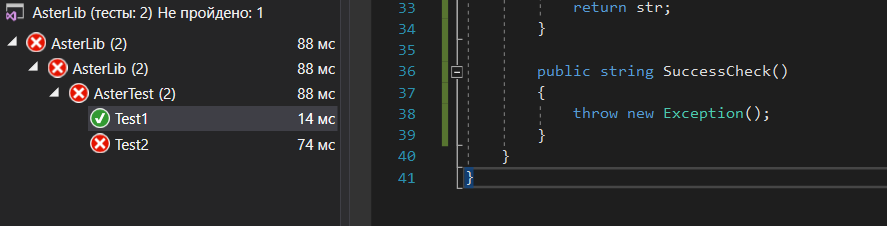


Рисунок 2 – Результаты тестирования с реализацией одного метода

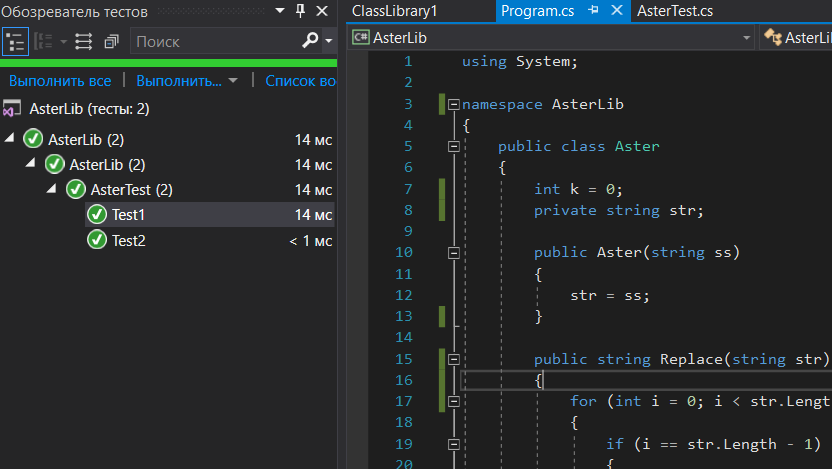


Рисунок 3 – Результаты тестирования готовой программы

ВЫВОД

В ходе выполнения лабораторной работы были получены практические навыки использования фреймворка NUnit для модульного тестирования программного обеспечения. Были разработаны тесты для тестирования программы замены символов.