Министерство образования и науки Российской Федерации

Севастопольский государственный университет

Кафедра ИС

Отчет

По дисциплине: “Тестирование программного обеспечения”

Лабораторная работа №3

“Исследование способов модульного тестирования

программного обеспечения”

Выполнил:

ст.гр. ИС/б-32

Волобуев Ю.С.

Проверила:

Тлуховская Н.П.

Севастополь

2019

1 ЦЕЛЬ РАБОТЫ

Исследовать основные подходы к модульному тестированию программного обеспечения. Приобрести практические навыки составления модульных тестов для объектно-ориентированных программ.

2 ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Выбрать в качестве тестируемого один из классов, спроектированных в лабораторных работах № 1 – 4.
2. Составить спецификацию тестового случая для одного из методов выбранного класса.
3. Реализовать тестируемый класс и необходимое тестовое окружение.
4. Выполнить тестирование с выводом результатов на экран и сохранением в log-файл.
5. Проанализировать результаты тестирования, сделать выводы

3 СПЕЦИФИКАЦИЯ ТЕСТОВОГО СЛУЧАЯ

Название тестируемого класса: “AsterCheck”.

Описание тестового случая: Тест проверяет hf,jne метода “Aster” – Поиск в строке символа ‘\*’ и замена следующего после него знака знаком ‘-’. Для теста были выбраны следующие строки

1. Первая строка,
2. \*Вторая строка,
3. Третья строка\*,
4. Четвертая строк\*а,
5. Пят\*ая ст\*рока,
6. Шест\*\*ая строка,
7. "\*\*\*\*\*\*\*\*"

4 ТЕКСТ ПРОГРАММЫ

using System;

using System.IO;

namespace ConsoleApp3

{

class Program

{

static void Main(string[] args)

{

\_ = new AsterTester();

Log.Close();

}

}

public class AsterCheck

{

public Tuple<string, string> Aster(string str)

{

int k = 0;

string ss = 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);

}

}

if (k > 0)

{

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

}

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

return new Tuple<string, string>(ss, str);

}

}

public class Log

{

private static readonly StreamWriter log = new StreamWriter("log.txt");

static public void AddString(string s)

{

log.WriteLine(s);

}

static public void Close()

{

log.Close();

}

}

abstract class Tester

{

protected void LogMessage(string s)

{

Log.AddString(s);

}

}

class AsterTester : Tester

{

private readonly AsterCheck OUT;

public AsterTester()

{

OUT = new AsterCheck();

Run();

}

private void Run()

{

AsterTest1();

}

public void AsterTest1()

{

string[] strings = {

"Первая строка",

"\*Вторая строка",

"Третья строка\*",

"Четвертая строк\*а",

"Пят\*ая ст\*рока",

"Шест\*\*ая строка",

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

};

foreach (var str in strings)

{

var result = OUT.Aster(str);

Log.AddString($"Исходная строка: {result.Item1}");

Log.AddString($"Конечная строка: {result.Item2} \n\n");

}

}

}

}

5 РЕЗУЛЬТАТЫ

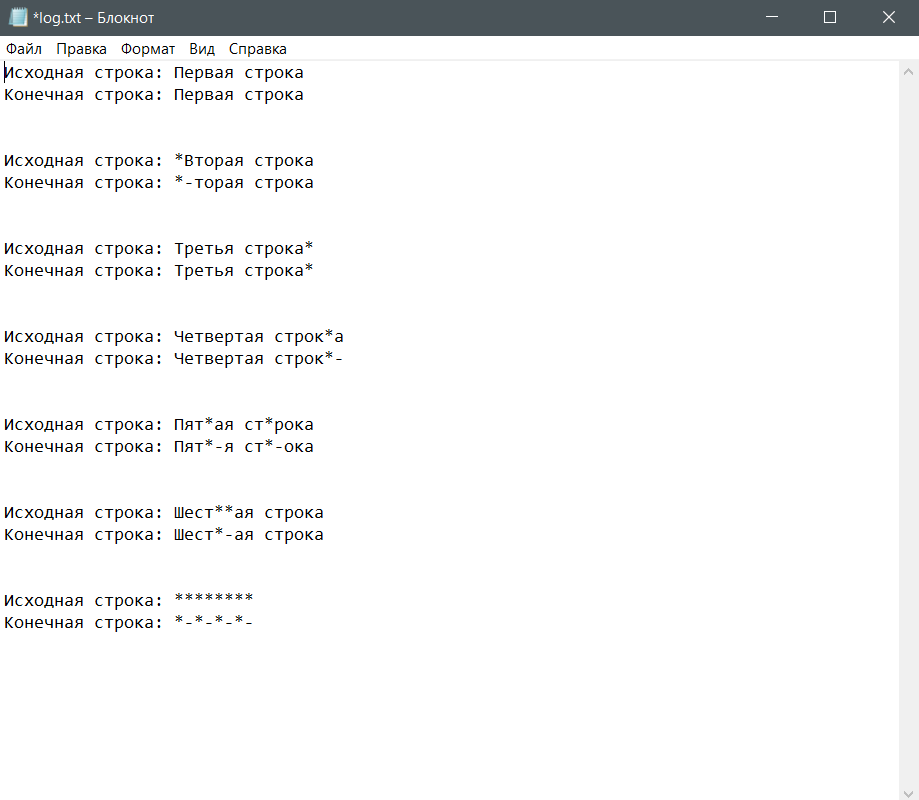


Рисунок 1 – Логи программы, сохраненные в файле

ВЫВОД

В ходе выполнения лабораторной работы были исследованы основные подходы к модульному тестированию программного обеспечения. Приобретены практические навыки составления модульных тестов для объектно-ориентированных программ.

Были усвоены шаги выполнения модульного тестирования и в итоге была написана программа, тестирующая модуль, выполняющий задания из Лаб. 1. Тестирование выполняется успешно и результаты, занесенные в лог соответствуют спецификации.