МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования

«Севастопольский государственный университет»

Институт информационных технологий и управления в технических системах

кафедра Информационные системы

09.03.02 Информационные системы и технологии (уровень бакалавриата)

Лабораторная работа №5

по дисциплине: «Тестирование программного обеспечения»

на тему: «Исследование способов модульного тестирования программного обеспечения в среде NUnit»

Вариант – 2

Выполнил

студент 3 курса группы ИС/б-33-о

Генералов Николай Николаевич

Отметка о зачете\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_

(дата)

Проверил

ст. пр. \_\_\_\_\_\_\_\_\_\_\_ Строганов В. А.

(должность) (подпись) (фамилия, инициалы)

г. Севастополь

2018 г.

# ЦЕЛЬ РАБОТЫ

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

# ПОСТАНОВКА ЗАДАЧИ

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

# ХОД РАБОТЫ

1. В качестве тестируемого класса был выбран класс MyMatrix из 1 задания лабораторной работы №1, который инициализируется двумерным массивом и производит подсчет не нулевых столбцов в нем.

Тестирование проводилось средствами стандартного фреймворка интегрированной среды разработки Visual Studio 2017 Community – XUnit.

В начале был составлен набор модульных тестов для каждого метода класса, а так же шаблон класса MyMatrix.

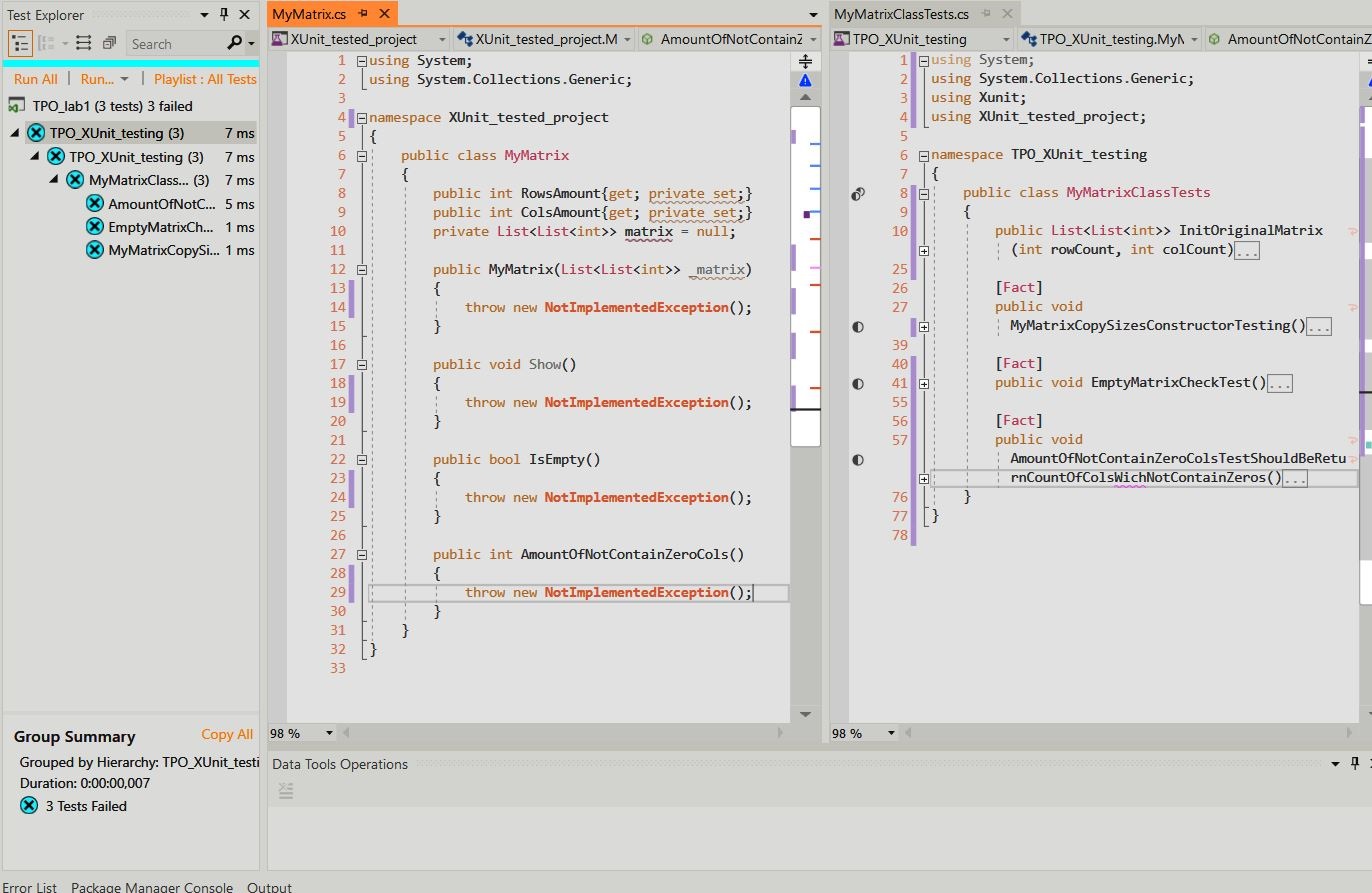


Рисунок 1 – Результат выполнения тестов не реализованных методов класса MyMatrix

Далее, осуществляя реализацию методов класса, тесты были запущенны повторно.

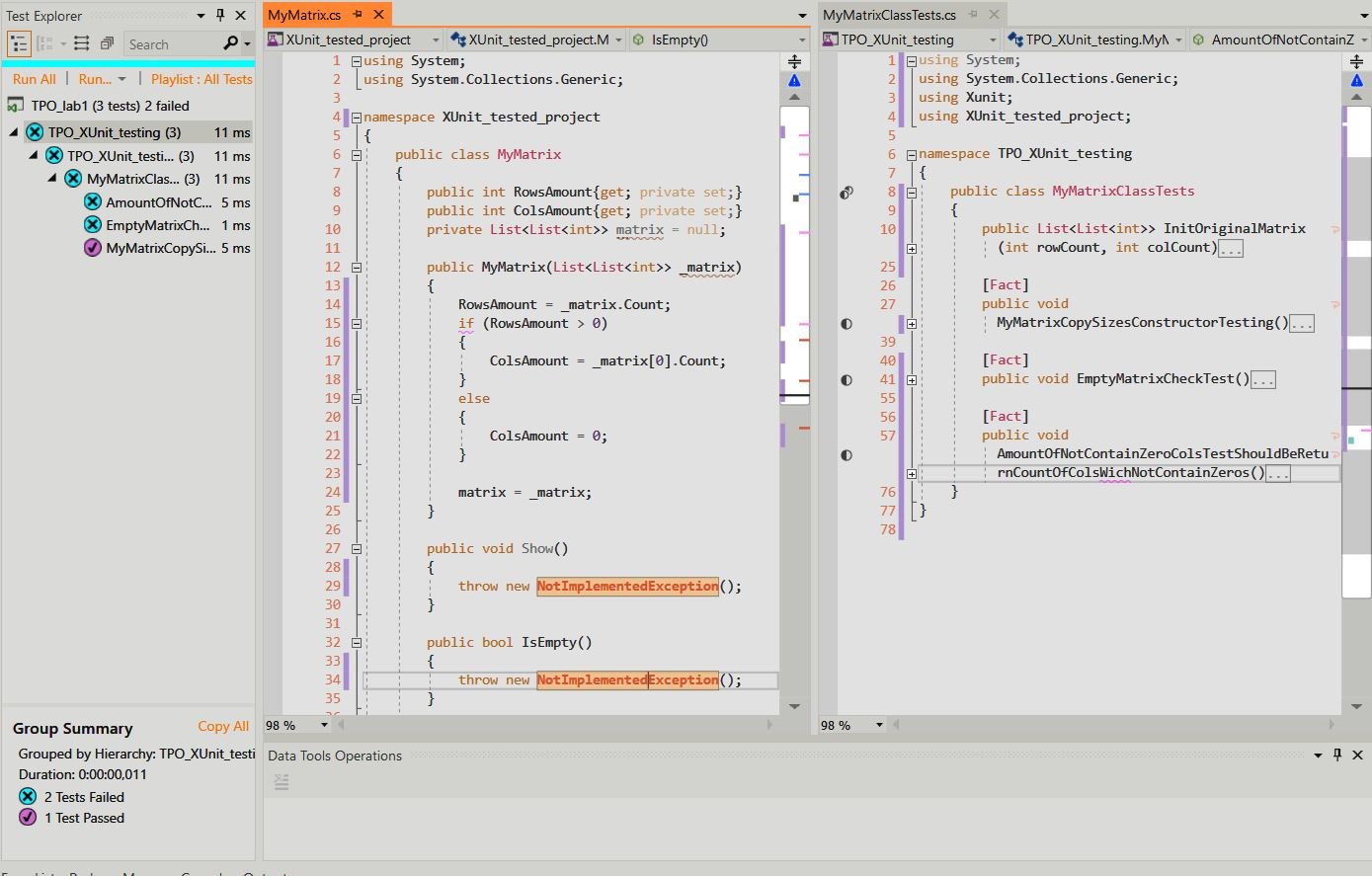


Рисунок 2 – Результат выполнения тестов с реализацией конструктора класса MyMatrix

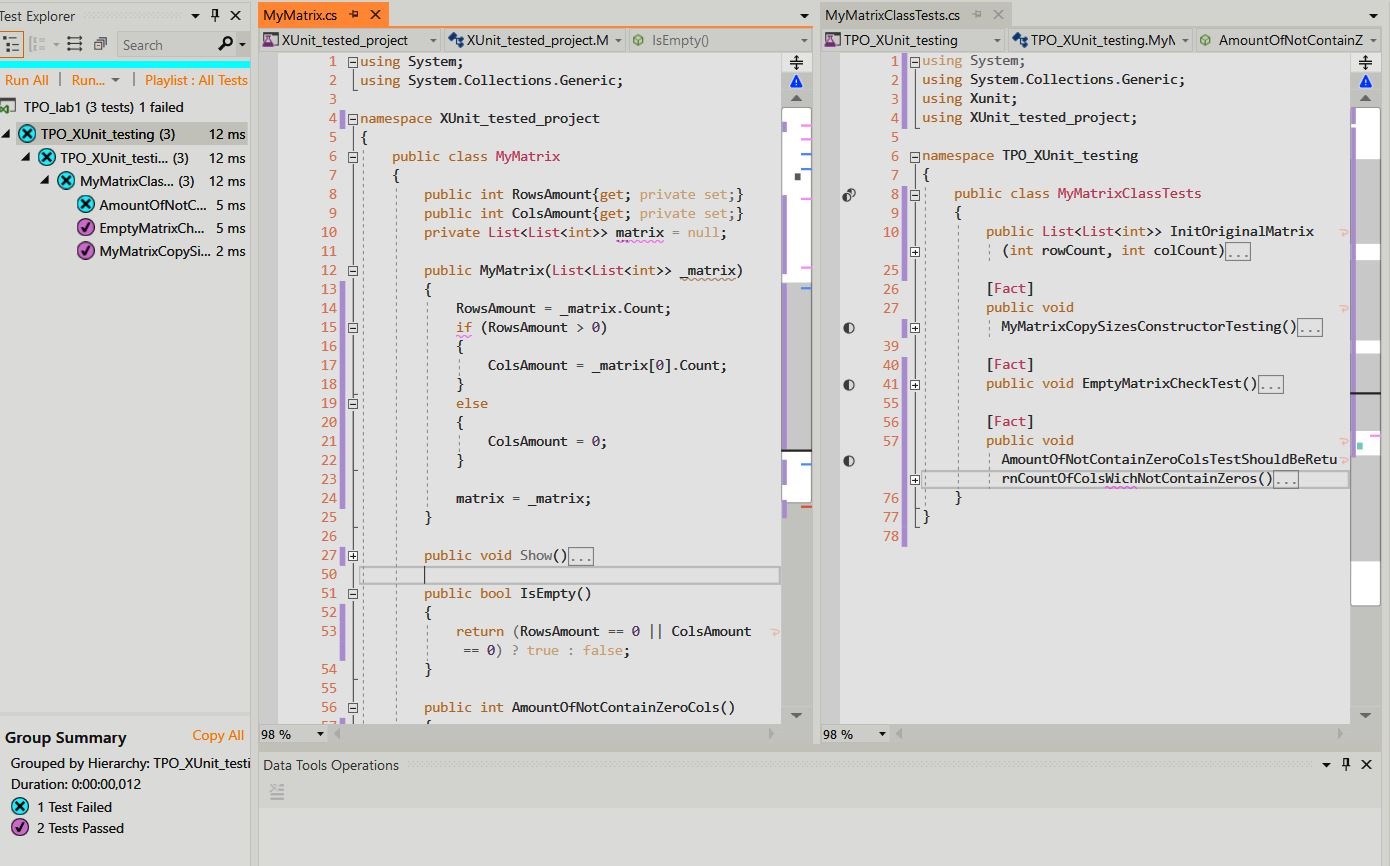


Рисунок 3 – Результат выполнения тестов с реализацией метода IsEmpty класса MyMatrix

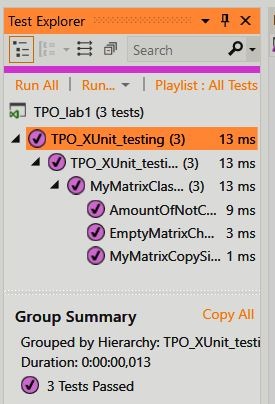


Рисунок 4 – Результат выполнения тестов при полной реализации класса MyMatrix

Так как весь набор тестов выполняется успешно, реализация класса MyMatrix была завершена.

# ТЕКСТ ПРОГРАММНЫХ МОДУЛЕЙ

Файл “MyMatrix.cs”

using System;

using System.Collections.Generic;

namespace XUnit\_tested\_project

{

public class MyMatrix

{

public int RowsAmount{get; private set;}

public int ColsAmount{get; private set;}

private List<List<int>> matrix = null;

public MyMatrix(List<List<int>> \_matrix)

{

RowsAmount = \_matrix.Count;

if (RowsAmount > 0)

{

ColsAmount = \_matrix[0].Count;

}

else

{

ColsAmount = 0;

}

matrix = \_matrix;

}

public void Show()

{

try

{

if (this.IsEmpty())

{

Console.WriteLine("Matrix is empty");

return;

}

foreach (var i in matrix)

{

foreach (var j in i)

{

Console.Write("{0, 2}" + " ", j);

}

Console.WriteLine();

}

}

catch (Exception err)

{

Console.WriteLine(err);

}

}

public bool IsEmpty()

{

return (RowsAmount == 0 || ColsAmount == 0) ? true : false;

}

public int AmountOfNotContainZeroCols()

{

try

{

int result = 0;

if (!this.IsEmpty())

{

for (int i = 0; i < this.ColsAmount; ++i)

{

bool ColContainZero = false;

for (int j = 0; j < this.RowsAmount; ++j)

{

if (matrix[j][i] == 0)

{

ColContainZero = true;

break;

}

}

if (!ColContainZero)

{

++result;

}

}

}

return result;

}

catch (Exception err)

{

Console.WriteLine(err);

return 0;

}

}

}

}

Файл “MyMatrixClassTests.cs”

using System;

using System.Collections.Generic;

using Xunit;

using XUnit\_tested\_project;

namespace TPO\_XUnit\_testing

{

public class MyMatrixClassTests

{

public List<List<int>> InitOriginalMatrix(int rowCount, int colCount)

{

List<List<int>> originalMatrix = new List<List<int>>();

for (int i = 0; i < rowCount; ++i)

{

originalMatrix.Add(new List<int>());

for (int j = 0; j < colCount; ++j)

{

originalMatrix[i].Add(i + j);

}

}

return originalMatrix;

}

[Fact]

public void MyMatrixCopySizesConstructorTesting()

{

//Arrange

List<List<int>> originalMatrix = InitOriginalMatrix(4, 4);

//Act

MyMatrix testingObject = new MyMatrix(originalMatrix);

//Assert

Assert.Equal(originalMatrix.Count, testingObject.RowsAmount);

Assert.Equal(originalMatrix[0].Count, testingObject.ColsAmount);

}

[Fact]

public void EmptyMatrixCheckTest()

{

//Arrange

List<List<int>> originalMatrix = InitOriginalMatrix(5, 0);

MyMatrix testingObject = new MyMatrix(originalMatrix);

//Act

var result = testingObject.IsEmpty();

var expectedVal = true;

//Assert

Assert.Equal(expectedVal, result);

}

[Fact]

public void AmountOfNotContainZeroColsTestShouldBeReturnCountOfColsWichNotContainZeros()

{

//Arrange

List<List<int>> originalMatrix = InitOriginalMatrix(5, 5);

MyMatrix testingObject = new MyMatrix(originalMatrix);

originalMatrix[0][0] = 0;

originalMatrix[0][3] = 0;

originalMatrix[0][4] = 0;

//Act

var result = testingObject.AmountOfNotContainZeroCols();

var expectedVal = 2;

//Assert

Assert.Equal(expectedVal, result);

}

}

}

# ВЫВОД

В ходе выполнения лабораторной работы были получены практические навыки тестирования классов при помощи фреймворков семейства XUnit, для модульного тестирования программного обеспечения.

При выполнении лабораторной работы согласно методологии разработки программного обеспечения TDD был разработан набор тестов, по результатам выполнения которых отслеживался процесс реализации выбранного класса. По результатам проведенных тестов, можно сделать вывод, что класс реализован верно и поставленная цель достигнута.