Министерство науки и высшего образования

Российской Федерации

Федеральное государственное бюджетное образовательное   
учреждение высшего образования

«Новосибирский государственный технический университет»



Кафедра теоретической и прикладной информатики

Лабораторная работа № 1  
по дисциплине «Метрология, качество и тестирование программного обеспечения»

**Модульное тестирование программ на языке C# средствами Visual Studio**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Факультет: | ПМИ |  |  |
| Группа: | ПМИМ-01 |  |  |
| Студент: | Ничипиенко В.А. |  |  |
| Вариант: | 6 |  |  |
| Преподаватель: | Зайцев М.Г. |  |  |

Новосибирск

2021

1. **Цель работы**

Сформировать практические навыки разработки модульных тестов для тестирования функций классов и выполнения модульного тестирования на языке С# с помощью средств автоматизации Visual Studio.

1. **Задание**

* Разработайте на языке С# класс, содержащий функции в соответствии с вариантом задания.
* Разработайте тестовые наборы данных по критерию С0.для тестирования функций класса.
* Протестируйте созданный класс с помощью средств автоматизации модульного тестирования Visual Studio.
* Напишите отчёт о результатах проделанной работы.

|  |  |
| --- | --- |
| Вариант № | Функция |
| 6(1) | Функция получает одномерный массив вещественных переменных. Вычисляет и возвращает произведение значений компонентов массива с нечетными значениями индексов. |
| Функция получает одномерный массив вещественных переменных и целое – параметр сдвига. Функция изменяет массив циклическим сдвигом значений его элементов вправо на число позиций, равное параметру сдвига. |
| Функция получает целое число b – основание системы счисления и строку s, содержащую представление дробной части числа в системы счисления с основанием b. Функция формирует и возвращает из строки s целое число. |

1. **Управляющий граф программы**

*Функция 1*

1. public static double CalculateTheProductOfOddArrayElements(double[] inputArray){
2. double answer = 1.0;
3. for (int i = 0; i < inputArray.Length; i++)
4. if (i % 2 != 0)
5. answer \*= inputArray[i];
6. return answer;}

*Функция 2*

1. public static double[] ShiftArrayElements(double[] inputArray, int shiftParameter) {
2. double[] newArray = new double[inputArray.Length];
3. for (int i = 0; i < inputArray.Length; i++)
4. newArray[(i + shiftParameter) % newArray.Length] = inputArray[i];
5. return newArray;}

*Функция 3*

1. public static int ConvertFractionalPartOfNumberToInteger(int b, string s){
2. string[] words = s.Split('.');
3. string fractionalPart = words[1];
4. double inputNumber = PFracTo10(fractionalPart, b);
5. double supportNumber = PIntTo10(Math.Pow(10, fractionalPart.Length).ToString(), b);
6. double answer = inputNumber \* supportNumber;
7. return Int10ToP((int)answer, b);}
8. **Тестовые наборы данных для тестирования функций класса**

*Функция 1*

|  |  |
| --- | --- |
| № теста | Входные данные |
| 1 | (1, 2, 3, 4, 5, 6, 7, 8, 9) |

*Функция 2*

|  |  |
| --- | --- |
| № теста | Входные данные |
| 1 | (1, 2, 3, 4, 5, 6, 7, 8, 9) |

*Функция 3*

|  |  |
| --- | --- |
| № теста | Входные данные |
| 1 | b = 8, s = “0.45” |

1. **Исходные тексты программ**

*Текст программы TestClass*

using System;

using System.Collections.Generic;

namespace Unit\_testing

{

public class TestClass

{

public static double CalculateTheProductOfOddArrayElements(double[] inputArray)

{

double answer = 1.0;

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

{

if (i % 2 != 0)

{

answer \*= inputArray[i];

}

}

return answer;

}

public static double[] ShiftArrayElements(double[] inputArray, int shiftParameter)

{

double[] newArray = new double[inputArray.Length];

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

{

newArray[(i + shiftParameter) % newArray.Length] = inputArray[i];

}

return newArray;

}

public static int ConvertFractionalPartOfNumberToInteger(int b, string s)

{

string[] words = s.Split('.');

string fractionalPart = words[1];

double inputNumber = PFracTo10(fractionalPart, b);

double supportNumber = PIntTo10(Math.Pow(10, fractionalPart.Length).ToString(), b);

double answer = inputNumber \* supportNumber;

return Int10ToP((int)answer, b);

}

private static int PIntTo10(string dec, int p)

{

double d = 0.0;

int num = 0;

for (var i = 0; i < dec.Length; i++)

{

d = (Convert.ToInt32(dec[i]) > 64 && Convert.ToInt32(dec[i]) < 71) ?

PCharToInt(dec[i]) : Convert.ToInt32(dec[i]) - 48;

num += Convert.ToInt32(Math.Pow(p, dec.Length - 1 - i) \* d);

}

return Convert.ToInt32(num);

}

private static double PFracTo10(string frac, int p)

{

double dig = 0.0;

double num = 0.0;

for (var i = 0; i < frac.Length; i++)

{

dig = (Convert.ToInt32(frac[i]) > 64 && Convert.ToInt32(frac[i]) < 71) ?

Convert.ToInt32(PCharToInt(frac[i])) : dig = Convert.ToInt32(frac[i]) - 48;

num += Math.Pow(p, -1 - i) \* dig;

}

return num;

}

private static int Int10ToP(int n, int p)

{

string res = "";

int x;

while (n != 0)

{

x = n % p;

if (x > 9)

res = IntToChar(x) + res;

else

res = x.ToString() + res;

n = n / p;

}

return Convert.ToInt32(res);

}

private static int PCharToInt(char sym)

{

switch (sym)

{

case 'A': return 10;

case 'B': return 11;

case 'C': return 12;

case 'D': return 13;

case 'E': return 14;

case 'F': return 15;

default: return -1; //Error

}

}

private static char IntToChar(int n)

{

Dictionary<int, char> dict = new Dictionary<int, char>()

{

{ 10, 'A' }, { 11, 'B' }, { 12, 'C' }, { 13, 'D' }, { 14, 'E' }, { 15, 'F' }

};

foreach (var e in dict)

if (e.Key == n)

return e.Value;

throw new ArgumentException("Error");

}

}

}

*Текст программы UnitTest*

using Microsoft.VisualStudio.TestTools.UnitTesting;

namespace UnitTestProject

{

[TestClass]

public class UnitTest

{

[TestMethod]

public void TestMethod1()

{

double[] args = new double[] { 1, 2, 3, 4, 5, 6, 7, 8, 9 };

double expectedResult = 384;

var result = Unit\_testing.TestClass.CalculateTheProductOfOddArrayElements(args);

Assert.AreEqual(expectedResult, result);

}

[TestMethod]

public void TestMethod2()

{

double[] args = new double[] { 1, 2, 3, 4, 5, 6, 7, 8, 9 };

double[] expectedResult = new double[] { 8, 9, 1, 2, 3, 4, 5, 6, 7 };

var result = Unit\_testing.TestClass.ShiftArrayElements(args, 2);

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

{

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

}

}

[TestMethod]

public void TestMethod3()

{

int b = 8;

string s = "0.45";

int expectedResult = 45;

var result = Unit\_testing.TestClass.ConvertFractionalPartOfNumberToInteger(b, s);

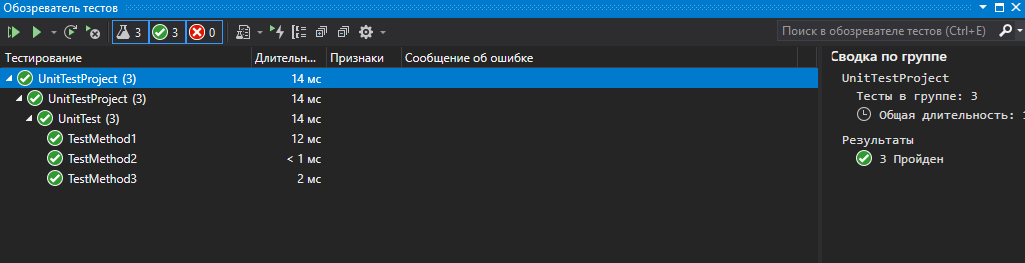
Assert.AreEqual(expectedResult, result);

}

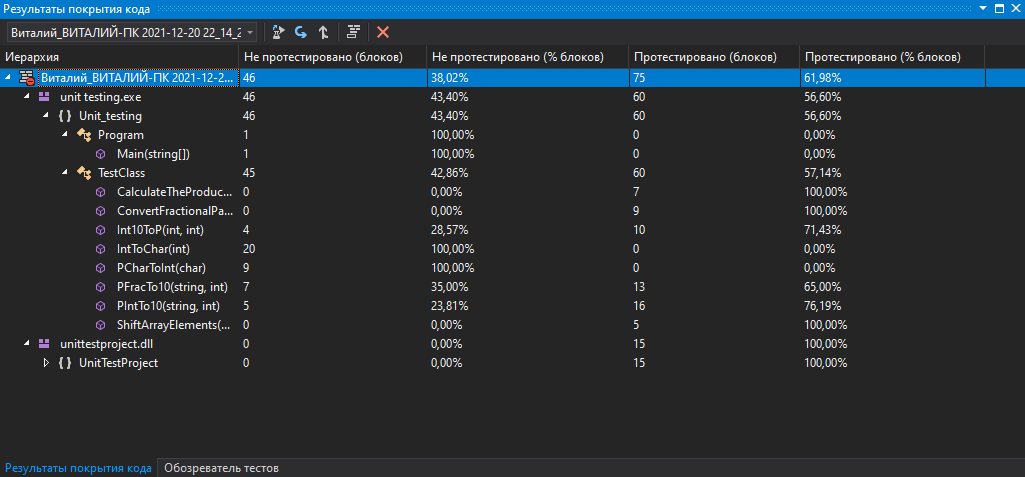
}

}

1. **Результаты выполнения модульных тестов**

****

1. **Результаты покрытия разработанного кода тестами**

****

1. **Выводы по выполненной работе**

В ходе данной работы были сформированы практические навыки разработки модульных тестов для тестирования функций классов и было выполнено модульное тестирование на языке С# с помощью средств автоматизации Visual Studio.