**Модульные тесты**

Пример модульного теста:

using Library;

using Microsoft.VisualStudio.TestTools.UnitTesting;

namespace UnitTestingsSamples1

{

[TestClass] // <= так нужно пометить класс, чтобы система тестирования начала искать в нем тесты.

public class QuadraticEquatonSolverTest

{

[TestMethod] // <= так нужно пометить метод, чтобы система тестирования поняла, что это тест.

public void TestSolve() // обратите внимание, что метод не статический.

{

// Arrange:

var a = 1;

var b = -3;

var c = 2;

//Act:

var result = QuadricEquation.Solve(a, b, c);

//Assert:

Assert.AreEqual(2, result.Length);

Assert.AreEqual(2, result[0]);

Assert.AreEqual(1, result[1]);

}

}

}

**Содержимое файла Program.cs (Equations)**

using Solver;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace umop7o3o3iLibraries

{

class Program

{

static void Main(string[] args)

{

var a = double.Parse(Console.ReadLine());

var b = double.Parse(Console.ReadLine());

var c = double.Parse(Console.ReadLine());

var result = QuadraticEquationsSolver.Solve(a, b, c);

Console.WriteLine(result[0]);

Console.WriteLine(result[1]);

Console.ReadKey();

}

}

}

**Содержимое файла QuadraticEquationsSolver.cs (Solver)**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Solver

{

public class QuadraticEquationsSolver

{

public static double[] Solve(double a, double b, double c)

{

var discriminant = b \* b - 4 \* a \* c;

var x1 = (-b + Math.Sqrt(discriminant)) / (2 \* a);

var x2 = (-b - Math.Sqrt(discriminant)) / (2 \* a);

return new[] { x1, x2 };

}

}

}

**Содержимое файла SolverTest.cs (Solver.Tests)**

using System;

using Microsoft.VisualStudio.TestTools.UnitTesting;

namespace Solver.Tests

{

[TestClass] // <= так нужно пометить класс, чтобы система тестирования начала искать в нем тесты.

public class QuadraticEquatonSolverTest

{

[TestMethod] // <= так нужно пометить метод, чтобы система тестирования поняла, что это тест.

public void OrdinaryEquations()

{

// Arrange:

var a = 1;

var b = -3;

var c = 2;

//Act:

var result = QuadraticEquationsSolver.Solve(a, b, c);

//Assert:

Assert.AreEqual(2, result.Length);

Assert.AreEqual(2, result[0]);

Assert.AreEqual(1, result[1]);

}

}

}