Midterm Practice

Name: Siyu Liu

Student ID: 8859412

# Question #1 – Test Case Development



# Question #2 – Test Execution / Defect Logging

## Screenshots

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with medium confidence

## Query

A screenshot of a computer

Description automatically generated

## Chart

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with medium confidence

# Question #3 – Unit Testing

## Source Code

### DetermineMinAndMax

namespace SiyuLiuMidtermPrework;

public class DetermineMinAndMax

{

public double valueOne;

public double valueTwo;

public double valueThree;

public string MaxValue()

{

var maxValue = Math.Max(Math.Max(valueOne, valueTwo), valueThree);

return $"The maximum value is {maxValue}";

}

public string MinValue()

{

var minValue = Math.Min(Math.Min(valueOne, valueTwo), valueThree);

return $"The minimum value is {minValue}";

}

// Generate Menu

public static bool Menu()

{

Console.WriteLine("### Input Three Values to Get the Maximum and Minimum ###");

var firstValue = NumberValidator("1. The First Value:");

if (!firstValue.isValid)

{

return false;

}

var secondValue = NumberValidator("2. The Second Value:");

if (!secondValue.isValid)

{

return false;

}

var thirdValue = NumberValidator("3. The Third Value:");

if (!thirdValue.isValid)

{

return false;

}

var determineMinAndMax = new DetermineMinAndMax()

{

valueOne = firstValue.validNumber,

valueTwo = secondValue.validNumber,

valueThree = thirdValue.validNumber

};

Console.WriteLine("### Result: ###");

Console.WriteLine(determineMinAndMax.MaxValue());

Console.WriteLine(determineMinAndMax.MinValue());

return true;

}

// Check whether user input is number

public static (bool isValid, double validNumber) NumberValidator(string OutputMessage)

{

Console.WriteLine(OutputMessage);

var inputValue = Console.ReadLine();

double validNumber;

bool isValid = Double.TryParse(inputValue, out validNumber);

if (isValid)

{

return (true, validNumber);

}

Console.WriteLine("The input value is not a valid number!");

return (false, 0);

}

}

### TestClass

using NUnit.Framework;

using SiyuLiuMidtermPrework;

[TestFixture]

public class TestClass

{

[Test]

public void MaxValue\_Input1and2and3\_ExpectMaxIs3() {

// Arrage

double value1 = 1;

double value2 = 2;

double value3 = 3;

string expected\_outcome = "The maximum value is 3";

// Act

var determineMinAndMax = new DetermineMinAndMax();

determineMinAndMax.valueOne = value1;

determineMinAndMax.valueTwo = value2;

determineMinAndMax.valueThree = value3;

var actual\_outcome = determineMinAndMax.MaxValue();

// Assert

Assert.AreEqual(expected\_outcome, actual\_outcome);

}

[Test]

public void MaxValue\_Input0andMinus1and1\_ExpectMaxIs1()

{

// Arrage

double value1 = 0;

double value2 = -1;

double value3 = 1;

string expected\_outcome = "The maximum value is 1";

// Act

var determineMinAndMax = new DetermineMinAndMax();

determineMinAndMax.valueOne = value1;

determineMinAndMax.valueTwo = value2;

determineMinAndMax.valueThree = value3;

var actual\_outcome = determineMinAndMax.MaxValue();

// Assert

Assert.AreEqual(expected\_outcome, actual\_outcome);

}

[Test]

public void MaxValue\_InputPoint5and1and2\_ExpectMaxIs2()

{

// Arrage

double value1 = 0.5;

double value2 = 1;

double value3 = 2;

string expected\_outcome = "The maximum value is 2";

// Act

var determineMinAndMax = new DetermineMinAndMax();

determineMinAndMax.valueOne = value1;

determineMinAndMax.valueTwo = value2;

determineMinAndMax.valueThree = value3;

var actual\_outcome = determineMinAndMax.MaxValue();

// Assert

Assert.AreEqual(expected\_outcome, actual\_outcome);

}

[Test]

public void MaxValue\_Input0and0and1\_ExpectMaxIs1()

{

// Arrage

double value1 = 0;

double value2 = 0;

double value3 = 1;

string expected\_outcome = "The maximum value is 1";

// Act

var determineMinAndMax = new DetermineMinAndMax();

determineMinAndMax.valueOne = value1;

determineMinAndMax.valueTwo = value2;

determineMinAndMax.valueThree = value3;

var actual\_outcome = determineMinAndMax.MaxValue();

// Assert

Assert.AreEqual(expected\_outcome, actual\_outcome);

}

[Test]

public void MaxValue\_Input0and2Point5and1\_ExpectMaxIs3()

{

// Arrage

double value1 = 0;

double value2 = 2.5;

double value3 = 1;

string expected\_outcome = "The maximum value is 2.5";

// Act

var determineMinAndMax = new DetermineMinAndMax();

determineMinAndMax.valueOne = value1;

determineMinAndMax.valueTwo = value2;

determineMinAndMax.valueThree = value3;

var actual\_outcome = determineMinAndMax.MaxValue();

// Assert

Assert.AreEqual(expected\_outcome, actual\_outcome);

}

[Test]

public void MaxValue\_InputMinus1andMinus2andMinus4\_ExpectMaxIsMinus1()

{

// Arrage

double value1 = -1;

double value2 = -2;

double value3 = -4;

string expected\_outcome = "The maximum value is -1";

// Act

var determineMinAndMax = new DetermineMinAndMax();

determineMinAndMax.valueOne = value1;

determineMinAndMax.valueTwo = value2;

determineMinAndMax.valueThree = value3;

var actual\_outcome = determineMinAndMax.MaxValue();

// Assert

Assert.AreEqual(expected\_outcome, actual\_outcome);

}

[Test]

public void MaxValue\_Input1Point1and1Point2and1Point3\_ExpectMaxIs1Point3()

{

// Arrage

double value1 = 1.1;

double value2 = 1.2;

double value3 = 1.3;

string expected\_outcome = "The maximum value is 1.3";

// Act

var determineMinAndMax = new DetermineMinAndMax();

determineMinAndMax.valueOne = value1;

determineMinAndMax.valueTwo = value2;

determineMinAndMax.valueThree = value3;

var actual\_outcome = determineMinAndMax.MaxValue();

// Assert

Assert.AreEqual(expected\_outcome, actual\_outcome);

}

[Test]

public void MinValue\_Input1and2and3\_ExpectMinIs1()

{

// Arrage

double value1 = 1;

double value2 = 2;

double value3 = 3;

string expected\_outcome = "The minimum value is 1";

// Act

var determineMinAndMax = new DetermineMinAndMax();

determineMinAndMax.valueOne = value1;

determineMinAndMax.valueTwo = value2;

determineMinAndMax.valueThree = value3;

var actual\_outcome = determineMinAndMax.MinValue();

// Assert

Assert.AreEqual(expected\_outcome, actual\_outcome);

}

[Test]

public void MinValue\_Input0and1and2\_ExpectMinIs0()

{

// Arrage

double value1 = 0;

double value2 = 1;

double value3 = 2;

string expected\_outcome = "The minimum value is 0";

// Act

var determineMinAndMax = new DetermineMinAndMax();

determineMinAndMax.valueOne = value1;

determineMinAndMax.valueTwo = value2;

determineMinAndMax.valueThree = value3;

var actual\_outcome = determineMinAndMax.MinValue();

// Assert

Assert.AreEqual(expected\_outcome, actual\_outcome);

}

[Test]

public void MinValue\_InputMinus1and0and1\_ExpectMinIsMinus1()

{

// Arrage

double value1 = -1;

double value2 = 0;

double value3 = 1;

string expected\_outcome = "The minimum value is -1";

// Act

var determineMinAndMax = new DetermineMinAndMax();

determineMinAndMax.valueOne = value1;

determineMinAndMax.valueTwo = value2;

determineMinAndMax.valueThree = value3;

var actual\_outcome = determineMinAndMax.MinValue();

// Assert

Assert.AreEqual(expected\_outcome, actual\_outcome);

}

[Test]

public void MinValue\_InputPoint5and1and2\_ExpectMinIsPoint5()

{

// Arrage

double value1 = 0.5;

double value2 = 1;

double value3 = 2;

string expected\_outcome = "The minimum value is 0.5";

// Act

var determineMinAndMax = new DetermineMinAndMax();

determineMinAndMax.valueOne = value1;

determineMinAndMax.valueTwo = value2;

determineMinAndMax.valueThree = value3;

var actual\_outcome = determineMinAndMax.MinValue();

// Assert

Assert.AreEqual(expected\_outcome, actual\_outcome);

}

[Test]

public void MinValue\_Input1and1and0\_ExpectMinIs0()

{

// Arrage

double value1 = 1;

double value2 = 1;

double value3 = 0;

string expected\_outcome = "The minimum value is 0";

// Act

var determineMinAndMax = new DetermineMinAndMax();

determineMinAndMax.valueOne = value1;

determineMinAndMax.valueTwo = value2;

determineMinAndMax.valueThree = value3;

var actual\_outcome = determineMinAndMax.MinValue();

// Assert

Assert.AreEqual(expected\_outcome, actual\_outcome);

}

[Test]

public void MinValue\_InputMinusPoint5and0and1\_ExpectMinIsMinusPoint5()

{

// Arrage

double value1 = -0.5;

double value2 = 0;

double value3 = 1;

string expected\_outcome = "The minimum value is -0.5";

// Act

var determineMinAndMax = new DetermineMinAndMax();

determineMinAndMax.valueOne = value1;

determineMinAndMax.valueTwo = value2;

determineMinAndMax.valueThree = value3;

var actual\_outcome = determineMinAndMax.MinValue();

// Assert

Assert.AreEqual(expected\_outcome, actual\_outcome);

}

[Test]

public void MinValue\_InputMinusPoint5andMinus1and2\_ExpectMinIsMinus1()

{

// Arrage

double value1 = -0.5;

double value2 = -1;

double value3 = 2;

string expected\_outcome = "The minimum value is -1";

// Act

var determineMinAndMax = new DetermineMinAndMax();

determineMinAndMax.valueOne = value1;

determineMinAndMax.valueTwo = value2;

determineMinAndMax.valueThree = value3;

var actual\_outcome = determineMinAndMax.MinValue();

// Assert

Assert.AreEqual(expected\_outcome, actual\_outcome);

}

}

## Screenshot

A screenshot of a computer

Description automatically generated