```
LinksPlatform's Platform.Incrementers Class Library
    ./csharp/Platform.Incrementers/IIncrementer.cs
   using System.Runtime.CompilerServices;
   namespace Platform. Incrementers
   {
4
        /// <summary>
5
       /// <para>Defines an incrementer that increments any object.</para>
       /// <para>Определяет инкрементер, который выполняет приращение любого объекта.</para>
       /// </summary>
       public interface IIncrementer
10
            /// <summary>
11
12
            /// <para>Increments the object by a specific value.</para>
            /// <para>Выполняет приращение объекта на определённое значение.</para>
13
            /// </summary>
14
            [MethodImpl(MethodImplOptions.AggressiveInlining)]
15
            void Increment();
       }
17
18
1.2
    ./csharp/Platform.Incrementers/IIncrementer[TNumber].cs
   using System.Runtime.CompilerServices;
3
   namespace Platform. Incrementers
4
        /// <summary>
5
       /// <para>Defines an incrementer that increments any number with a specific type.</para>
6
       /// <para>Определяет инкрементер, который выполняет приращение любого числа определенного
          типа.</para>
        /// </summary>
       /// <typeparam name="TNumber"><para>Туре of incremented number.</para><para>Тип
9
           увеличиваемого числа.</para></typeparam>
10
       public interface IIncrementer<TNumber>
11
            /// <summary>
12
            /// <para>Increments the number by a specific value.</para>
13
            /// <para>Увеличивает число на определённое значение.</para>
            /// </summary>
15
            /// <param name="number"><para>The number to be incremented.</para><para>Увеличиваемое
16
                число.</para></param>
            /// <returns><para>The incremented number.</para><para>Увеличенное
17
               число.</para></returns>
            [MethodImpl(MethodImplOptions.AggressiveInlining)]
            TNumber Increment(TNumber number);
19
       }
20
   }
     ./csharp/Platform.Incrementers/Incrementer.cs
   using System.Runtime.CompilerServices;
   #pragma warning disable CS1591 // Missing XML comment for publicly visible type or member
4
   namespace Platform.Incrementers
6
        /// <remarks>
        /// Must be class, not struct (in order to persist access to Result field value).
       /// </remarks>
9
       public class Incrementer: IIncrementer
10
11
           protected ulong _result;
12
13
            public ulong Result
14
15
                [MethodImpl(MethodImplOptions.AggressiveInlining)]
16
17
                get => _result;
18
19
            [MethodImpl(MethodImplOptions.AggressiveInlining)]
20
           public Incrementer(ulong initialValue) => _result = initialValue;
21
22
23
            [MethodImpl(MethodImplOptions.AggressiveInlining)]
           public Incrementer() { }
25
            [MethodImpl(MethodImplOptions.AggressiveInlining)]
26
27
           public void Increment() => _result++;
       }
28
   }
29
```

```
./csharp/Platform.Incrementers/Incrementer[TValue, TDecision].cs
   using System.Runtime.CompilerServices;
   #pragma warning disable IDE0060 // Remove unused parameter
   #pragma warning disable CS1591 // Missing XML comment for publicly visible type or member
   namespace Platform. Incrementers
6
   {
        public class Incrementer<TValue, TDecision> : Incrementer
9
            private readonly TDecision _trueValue;
10
11
            [MethodImpl(MethodImplOptions.AggressiveInlining)]
            public Incrementer(ulong initialValue, TDecision trueValue) : base(initialValue) =>
13

    _trueValue = trueValue;

            [MethodImpl(MethodImplOptions.AggressiveInlining)]
15
            public Incrementer(TDecision trueValue) => _trueValue = trueValue;
16
17
            [MethodImpl(MethodImplOptions.AggressiveInlining)]
18
            public Incrementer() { }
20
            [MethodImpl(MethodImplOptions.AggressiveInlining)]
            public TDecision IncrementAndReturnTrue()
22
23
                _result++;
24
                return _trueValue;
25
            }
27
            [MethodImpl(MethodImplOptions.AggressiveInlining)]
            public TDecision IncrementAndReturnTrue(TValue value)
29
30
                _result++;
                return _trueValue;
32
            }
33
       }
34
35
    ./csharp/Platform.Incrementers/Incrementer[TValue].cs
1.5
   using System.Runtime.CompilerServices;
   #pragma warning disable CS1591 // Missing XML comment for publicly visible type or member
3
   namespace Platform. Incrementers
5
   {
6
        public class Incrementer<TValue> : Incrementer<TValue, bool>
            [MethodImpl(MethodImplOptions.AggressiveInlining)]
            public Incrementer(ulong initialValue) : base(initialValue, true) { }
10
11
12
            [MethodImpl(MethodImplOptions.AggressiveInlining)]
13
            public Incrementer() : base(true) { }
14
   }
15
    ./csharp/Platform.Incrementers.Tests/IncrementerTests.cs
1.6
   using Xunit;
   namespace Platform. Incrementers. Tests
        public class IncrementerTests
5
            [Fact]
            public void ParameterlessConstructedSetterTest()
                Incrementer<int> incrementer = new Incrementer<int>();
11
                Assert.Equal(default, incrementer.Result);
12
13
            [Fact]
14
            public void ConstructedWithDefaultValueSetterTest()
                Incrementer<int> incrementer = new Incrementer<int>(9UL);
17
                Assert.Equal(9UL, incrementer.Result);
18
            }
20
            [Fact]
21
            public void MethodsWithBooleanReturnTypeTest()
23
                Incrementer<int> incrementer = new Incrementer<int>();
```

```
incrementer.Increment();
25
                Assert.Equal(1UL, incrementer.Result);
                Assert.True(incrementer.IncrementAndReturnTrue());
^{27}
                Assert.Equal(2UL, incrementer.Result);
28
            }
30
            [Fact]
           public void MethodsWithIntegerReturnTypeTest()
{
31
^{32}
33
                Incrementer<int, int> incrementer = new Incrementer<int, int>(1);
34
                incrementer.Increment();
35
                Assert.Equal(1UL, incrementer.Result);
                Assert.Equal(1, incrementer.IncrementAndReturnTrue());
37
38
                Assert.Equal(2UL, incrementer.Result);
            }
       }
40
   }
41
```

Index

- ./csharp/Platform.Incrementers.Tests/IncrementerTests.cs, 2
 ./csharp/Platform.Incrementers/Ilncrementer.cs, 1
 ./csharp/Platform.Incrementers/Ilncrementer[TNumber].cs, 1
 ./csharp/Platform.Incrementers/Incrementer.cs, 1
 ./csharp/Platform.Incrementers/Incrementer[TValue, TDecision].cs, 1
 ./csharp/Platform.Incrementers/Incrementer[TValue].cs, 2