Komponentowe systemy rozproszone 2024/25

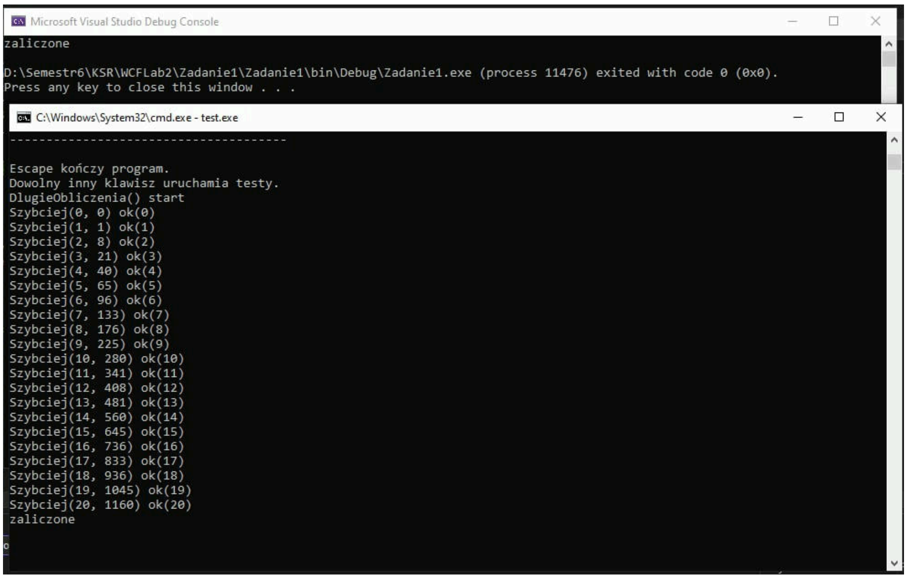
Hubert Wajda 193511

Wersja 1.0

Spis treści

1. Zadanie 1

1.1. Wynik

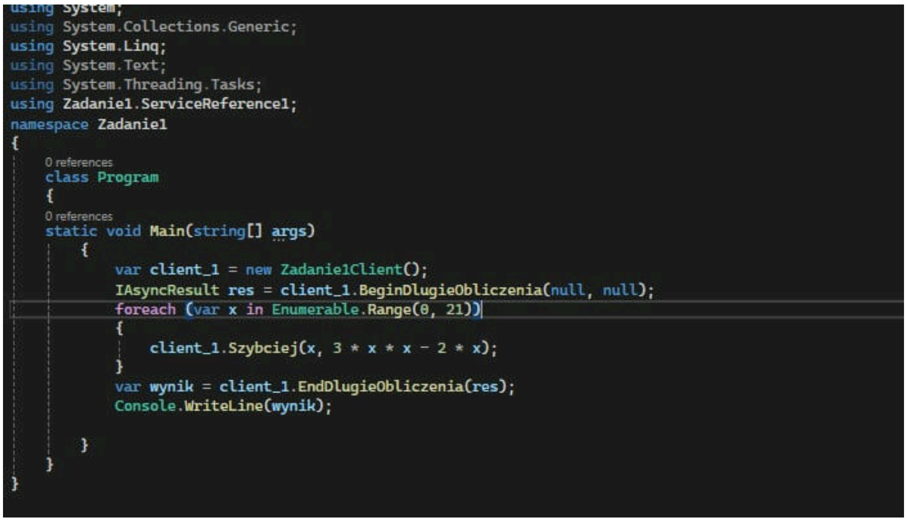

Microsoft Visual Studio Debug Console
C:\Windows\System32\cmd.exe - test.exe

```
zaliczone

D:\Semestr6\KSR\WCFLab2\Zadanie1\Zadanie1\bin\Debug\Zadanie1.exe (process 11476) exited with code 0 (0x0).
Press any key to close this window . . .
```

```
-------------------------------------

Escape kończy program.
Dowolny inny klawisz uruchamia testy.
DlugieObliczenia() start
Szybciej(0, 0) ok(0)
Szybciej(1, 1) ok(1)
Szybciej(2, 8) ok(2)
Szybciej(3, 21) ok(3)
Szybciej(4, 40) ok(4)
Szybciej(5, 65) ok(5)
Szybciej(6, 96) ok(6)
Szybciej(7, 133) ok(7)
Szybciej(8, 176) ok(8)
Szybciej(9, 225) ok(9)
Szybciej(10, 280) ok(10)
Szybciej(11, 341) ok(11)
Szybciej(12, 408) ok(12)
Szybciej(13, 481) ok(13)
Szybciej(14, 560) ok(14)
Szybciej(15, 645) ok(15)
Szybciej(16, 736) ok(16)
Szybciej(17, 833) ok(17)
Szybciej(18, 936) ok(18)
Szybciej(19, 1045) ok(19)
Szybciej(20, 1160) ok(20)
zaliczone
```

Figure 1: Zadanie 1 - Rezultat

1.2. Kod



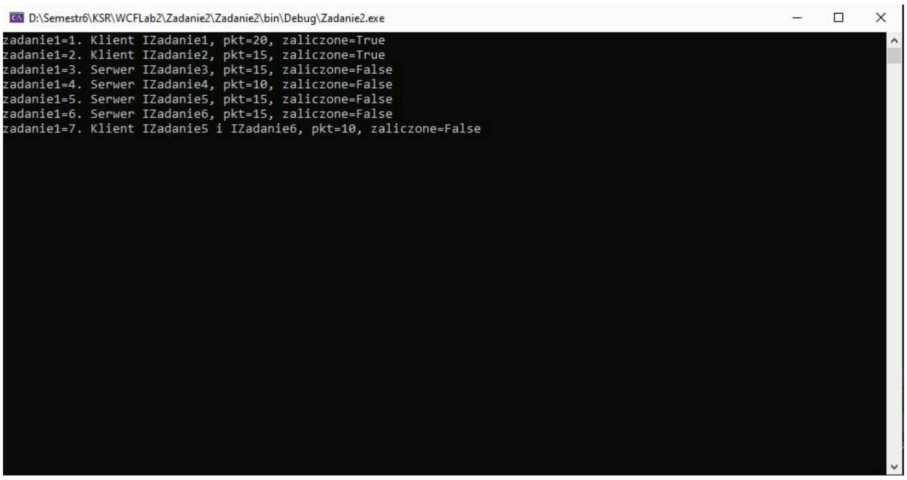
```csharp
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Zadanie1.ServiceReference1;
namespace Zadanie1
{
    class Program
    {
    static void Main(string[] args)
        {
            var client_1 = new Zadanie1Client();
            IAsyncResult res = client_1.BeginDlugieObliczenia(null, null);
            foreach (var x in Enumerable.Range(0, 21))
            {
                client_1.Szybciej(x, 3 * x * x - 2 * x);
            }
            var wynik = client_1.EndDlugieObliczenia(res);
            Console.WriteLine(wynik);

        }
    }
}
```

Figure 2: Zadanie 1 - Kod (Zadanie wykonane wykorzustując opcję „Generate asynchronous operations”)

```
namespace Zadanie1
{
    class Program
    {
    static void Main(string[] args)
        {
            var client_1 = new Zadanie1Client();
            IAsyncResult res = client_1.BeginDlugieObliczenia(null, null);
            foreach (var x in Enumerable.Range(0, 21))
            {
                client_1.Szybciej(x, 3 * x * x - 2 * x);
            }
            var wynik = client_1.EndDlugieObliczenia(res);
            Console.WriteLine(wynik);
            Console.ReadKey();

        }
    }
}
```

2. Zadanie 2

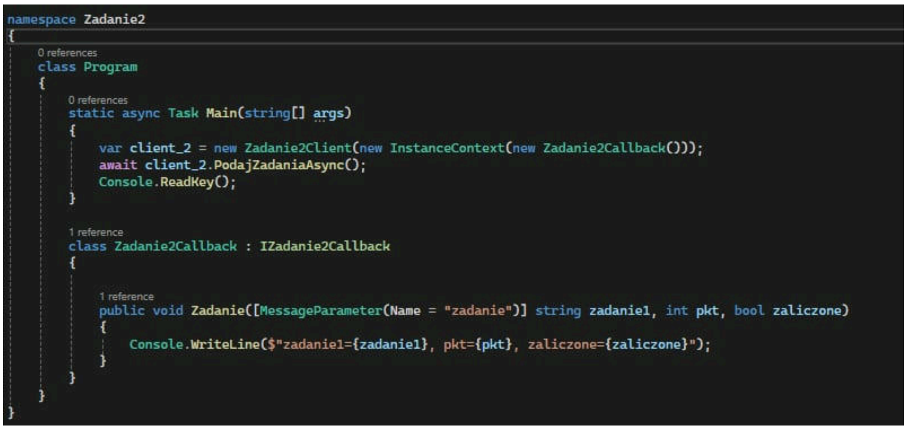
2.1. Wynik


D:\Semestr6\KSR\WCFLab2\Zadanie2\Zadanie2\bin\Debug\Zadanie2.exe

```
zadanie1=1. Klient IZadanie1, pkt=20, zaliczone=True
zadanie1=2. Klient IZadanie2, pkt=15, zaliczone=True
zadanie1=3. Serwer IZadanie3, pkt=15, zaliczone=False
zadanie1=4. Serwer IZadanie4, pkt=10, zaliczone=False
zadanie1=5. Serwer IZadanie5, pkt=15, zaliczone=False
zadanie1=6. Serwer IZadanie6, pkt=15, zaliczone=False
zadanie1=7. Klient IZadanie5 i IZadanie6, pkt=10, zaliczone=False
```

Figure 3: Zadanie 1 - Rezultat

2.2. Kod



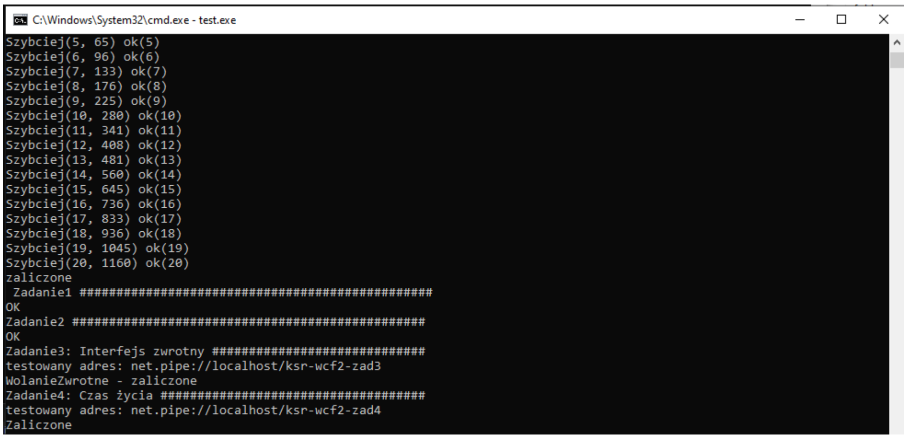
```
namespace Zadanie2
{
    0 references
    class Program
    {
        0 references
        static async Task Main(string[] args)
        {
            var client_2 = new Zadanie2Client(new InstanceContext(new Zadanie2Callback()));
            await client_2.PodajZadaniaAsync();
            Console.ReadKey();
        }

        1 reference
        class Zadanie2Callback : IZadanie2Callback
        {

            1 reference
            public void Zadanie([MessageParameter(Name = "zadanie")] string zadanie1, int pkt, bool zaliczone)
            {
                Console.WriteLine($"zadanie1={zadanie1}, pkt={pkt}, zaliczone={zaliczone}");
            }
        }
    }
}
```

Figure 4: Zadanie 1 - Kod (Zadanie wykonane wykorzustując opcję „Generate task-based operations”)

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Zadanie2.ServiceReference1;
using System.ServiceModel;

namespace Zadanie2
{
    class Program
    {
        static async Task Main(string[] args)
        {
            var client_2 = new Zadanie2Client(new InstanceContext(new
Zadanie2Callback()));
            await client_2.PodajZadaniaAsync();
            Console.ReadKey();
        }

        class Zadanie2Callback : IZadanie2Callback
        {

            public void Zadanie([MessageParameter(Name = "zadanie")] string zadanie1, int
pkt, bool zaliczone)
            {
                Console.WriteLine($"zadanie1={zadanie1}, pkt={pkt},
zaliczone={zaliczone}");
            }
        }
    }
}
```

3. Zadanie 3 i 4

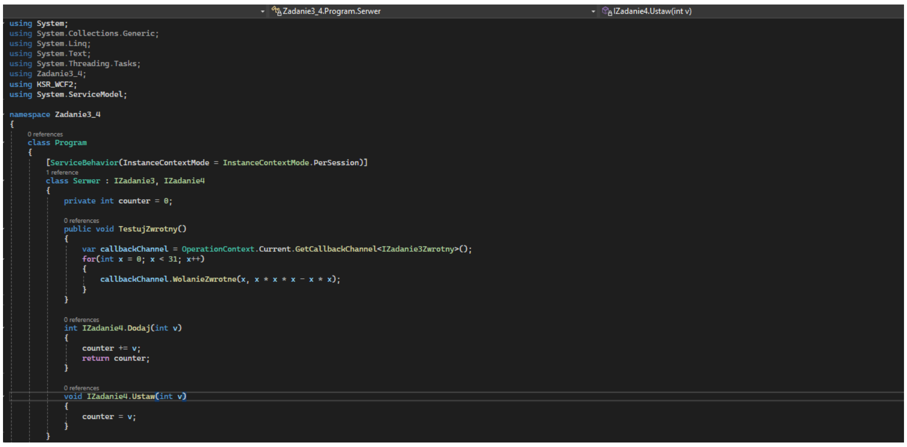
3.1. Wynik



```
Szybciej(5, 65) ok(5)
Szybciej(6, 96) ok(6)
Szybciej(7, 133) ok(7)
Szybciej(8, 176) ok(8)
Szybciej(9, 225) ok(9)
Szybciej(10, 280) ok(10)
Szybciej(11, 341) ok(11)
Szybciej(12, 408) ok(12)
Szybciej(13, 481) ok(13)
Szybciej(14, 560) ok(14)
Szybciej(15, 645) ok(15)
Szybciej(16, 736) ok(16)
Szybciej(17, 833) ok(17)
Szybciej(18, 936) ok(18)
Szybciej(19, 1045) ok(19)
Szybciej(20, 1160) ok(20)
zaliczone
 Zadanie1 ################################################
OK
Zadanie2 ################################################
OK
Zadanie3: Interfejs zwrotny #############################
testowany adres: net.pipe://localhost/ksr-wcf2-zad3
WolanieZwrotne - zaliczone
Zadanie4: Czas życia ####################################
testowany adres: net.pipe://localhost/ksr-wcf2-zad4
Zaliczone
```

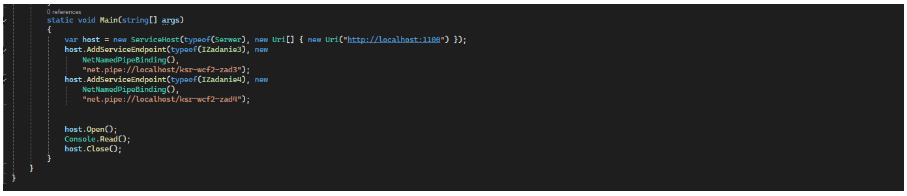
Figure 5: Zadanie 3 i 4 - Rezultat

3.2. Kod



```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Zadanie3_4;
using KSR_WCF2;
using System.ServiceModel;

namespace Zadanie3_4
{
    0 references
    class Program
    {
        [ServiceBehavior(InstanceContextMode = InstanceContextMode.PerSession)]
        1 reference
        class Serwer : IZadanie3, IZadanie4
        {
            private int counter = 0;

            0 references
            public void TestujZwrotny()
            {
                var callbackChannel = OperationContext.Current.GetCallbackChannel<IZadanie3Zwrotny>();
                for(int x = 0; x < 31; x++)
                {
                    callbackChannel.WolanieZwrotne(x, x * x * x - x * x);
                }
            }

            0 references
            int IZadanie4.Dodaj(int v)
            {
                counter += v;
                return counter;
            }

            0 references
            void IZadanie4.Ustaw(int v)
            {
                counter = v;
            }
        }
```

Figure 6: Zadanie 3 i 4 - Kod (część 1)



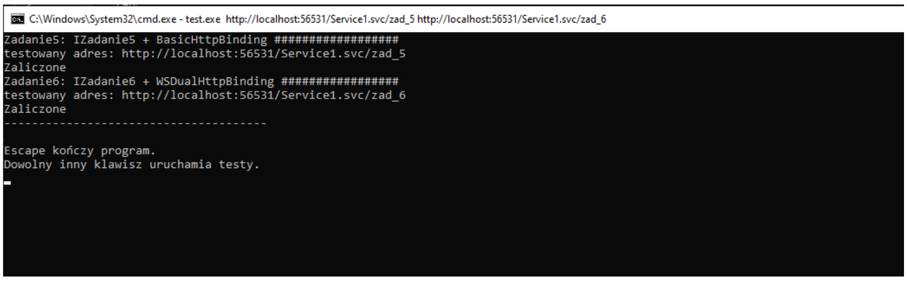
```
        0 references
        static void Main(string[] args)
        {
            var host = new ServiceHost(typeof(Serwer), new Uri[] { new Uri("http://localhost:1100") });
            host.AddServiceEndpoint(typeof(IZadanie3), new
                NetNamedPipeBinding(),
                "net.pipe://localhost/ksr-wcf2-zad3");
            host.AddServiceEndpoint(typeof(IZadanie4), new
                NetNamedPipeBinding(),
                "net.pipe://localhost/ksr-wcf2-zad4");


            host.Open();
            Console.Read();
            host.Close();
        }
    }
}
```

Figure 7: Zadanie 3 i 4 - Kod (część 2)

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Zadanie3_4;
using KSR_WCF2;
using System.ServiceModel;

namespace Zadanie3_4
{
    class Program
    {
        [ServiceBehavior(InstanceContextMode = InstanceContextMode.PerSession)]
        class Serwer : IZadanie3, IZadanie4
        {
            private int counter = 0;

            public void TestujZwrotny()
            {
                var callbackChannel =
OperationContext.Current.GetCallbackChannel<IZadanie3Zwrotny>();
                for(int x = 0; x < 31; x++)
                {
                    callbackChannel.WolanieZwrotne(x, x * x * x - x * x);
                }
            }

            int IZadanie4.Dodaj(int v)
            {
                counter += v;
                return counter;
            }

            void IZadanie4.Ustaw(int v)
            {
                counter = v;
            }
        }
        static void Main(string[] args)
        {
            var host = new ServiceHost(typeof(Serwer), new Uri[] { new Uri("http://
localhost:1100") });
            host.AddServiceEndpoint(typeof(IZadanie3), new
                NetNamedPipeBinding(),
                "net.pipe://localhost/ksr-wcf2-zad3");
            host.AddServiceEndpoint(typeof(IZadanie4), new
                NetNamedPipeBinding(),
                "net.pipe://localhost/ksr-wcf2-zad4");


            host.Open();
            Console.Read();
            host.Close();
        }
    }
}
```

4. Zadanie 5 i 6

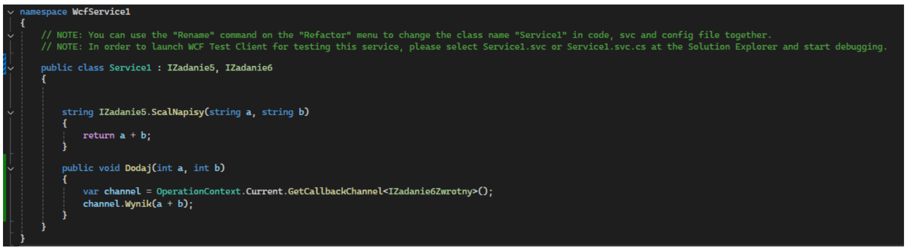
4.1. Wynik


C:\Windows\System32\cmd.exe - test.exe http://localhost:56531/Service1.svc/zad_5 http://localhost:56531/Service1.svc/zad_6

```
Zadanie5: IZadanie5 + BasicHttpBinding ##################
testowany adres: http://localhost:56531/Service1.svc/zad_5
Zaliczone
Zadanie6: IZadanie6 + WSDualHttpBinding #################
testowany adres: http://localhost:56531/Service1.svc/zad_6
Zaliczone
--------------------------------------

Escape kończy program.
Dowolny inny klawisz uruchamia testy.
_
```

Figure 8: Zadanie 5 i 6 - Rezultat

4.2. Kod



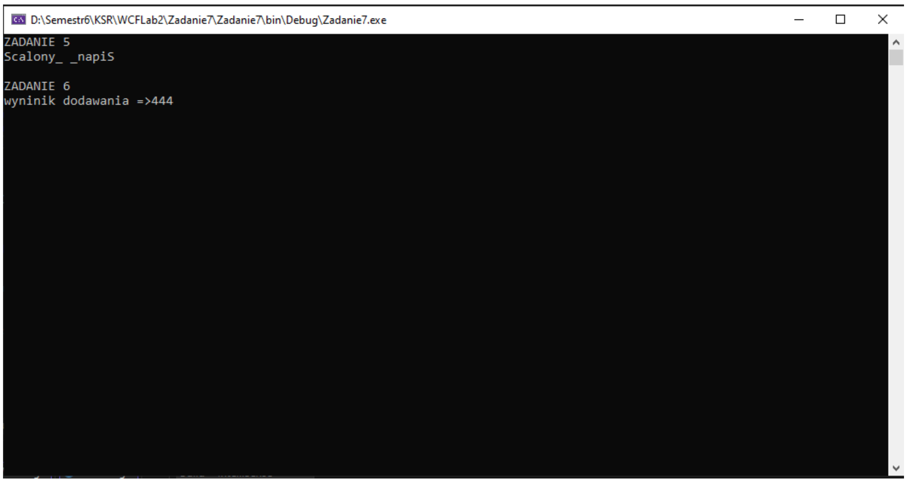
```
namespace WcfService1
{
    // NOTE: You can use the "Rename" command on the "Refactor" menu to change the class name "Service1" in code, svc and config file together.
    // NOTE: In order to launch WCF Test Client for testing this service, please select Service1.svc or Service1.svc.cs at the Solution Explorer and start debugging.

    public class Service1 : IZadanie5, IZadanie6
    {


        string IZadanie5.ScalNapisy(string a, string b)
        {
            return a + b;
        }

        public void Dodaj(int a, int b)
        {
            var channel = OperationContext.Current.GetCallbackChannel<IZadanie6Zwrotny>();
            channel.Wynik(a + b);
        }
    }
}
```

Figure 9: Zadanie 5 i 6 - Kod

```
namespace WcfService1
{
  // NOTE: You can use the "Rename" command on the "Refactor" menu to change the class
name "Service1" in code, svc and config file together.
  // NOTE: In order to launch WCF Test Client for testing this service, please select
Service1.svc or Service1.svc.cs at the Solution Explorer and start debugging.
  public class Service1 : IZadanie5, IZadanie6
  {

        string IZadanie5.ScalNapisy(string a, string b)
        {
            return a + b;
        }
        public void Dodaj(int a, int b)
        {
            var channel = OperationContext.Current.GetCallbackChannel<IZadanie6Zwrotny>();
            channel.Wynik(a + b);
        }
    }
}
```

5. Zadanie 7

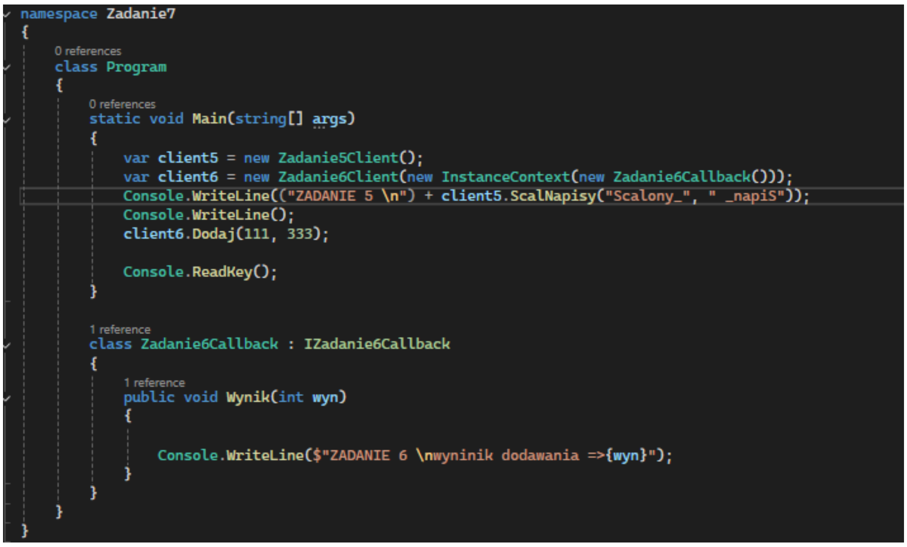
5.1. Wynik


D:\Semestr6\KSR\WCFLab2\Zadanie7\Zadanie7\bin\Debug\Zadanie7.exe

```
ZADANIE 5
Scalony_ _napiS

ZADANIE 6
wyninik dodawania =>444
```

Figure 10: Zadanie 7 - Rezultat

5.2. Kod



```
namespace Zadanie7
{
    0 references
    class Program
    {
        0 references
        static void Main(string[] args)
        {
            var client5 = new Zadanie5Client();
            var client6 = new Zadanie6Client(new InstanceContext(new Zadanie6Callback()));
            Console.WriteLine(("ZADANIE 5 \n") + client5.ScalNapisy("Scalony_", " _napiS"));
            Console.WriteLine();
            client6.Dodaj(111, 333);

            Console.ReadKey();
        }

        1 reference
        class Zadanie6Callback : IZadanie6Callback
        {
            1 reference
            public void Wynik(int wyn)
            {

                Console.WriteLine($"ZADANIE 6 \nwyninik dodawania =>{wyn}");
            }
        }
    }
}
```

Figure 11: Zadanie 7 - Kod

```
namespace Zadanie7
{
    class Program
    {
        static void Main(string[] args)
        {
            var client5 = new Zadanie5Client();
            var client6 = new Zadanie6Client(new InstanceContext(new Zadanie6Callback()));
            Console.WriteLine(("ZADANIE 5 \n") + client5.ScalNapisy("Scalony_", "
_napiS"));
            Console.WriteLine();
            client6.Dodaj(111, 333);

            Console.ReadKey();
        }

        class Zadanie6Callback : IZadanie6Callback
        {
            public void Wynik(int wyn)
            {

                Console.WriteLine($"ZADANIE 6 \nwyninik dodawania =>{wyn}");
            }
        }
    }
}
```