Wykład 4
Delegat (delegate), właściwości indeksowane, zdarzenie (event)

Zofia Kruczkiewicz

Zagadnienia

1. **Delegaty wiązane, właściwości indeksowane**
2. **Delegaty niewiązane**
3. **Nowa wersja kalkulatora, delegaty**
4. **Zastosowanie delegatów do obsługi zdarzeń użytkownika**

1. Delegaty wiązane (1)

Delegaty służą do sygnalizowania zdarzenia.

Definicja przykładowego delegatu:

public delegate void lista_Operacji(wchar_t oper_);

• typ referencyjny delegatu lista_Operacji (pochodny klasy System::Delegate)

• tworzona jest lista referencji do metod, które spełniają następujące warunki: nagłówek metody powinien zawierać parametr typu **wchar_t**, a zwracany wynik jest równy **void**.

lista_Operacji ^ lista_operacji=
gcnew lista_Operacji(kalkulator,&Dzialania_kalkulatora::obsluga);

• Metody mogą być typu **static** lub obiektowe.

Delegaty wiązane (2)

• **Utworzenie listy z jedną metodą obiektową**

```
lista_Operacji ^ lista_operacji=
        gcnew lista_Operacji (kalkulator,&Dzialania_kalkulatora::obsluga);
```

• **Dodanie do listy metody obiektową**

```
lista_operacji +=
        gcnew lista_Operacji (kalkulator,&Dzialania_kalkulatora::info2);
```

• **Usunięcie z listy metody obiektową**

```
lista_operacji -=
        gcnew lista_Operacji (kalkulator,&Dzialania_kalkulatora::info2);
```

•**Dodanie do listy metody statycznej**

```
lista_operacji += gcnew lista_Operacji (Dzialania_kalkulatora::info4);
```

Delegaty wiązane (3)

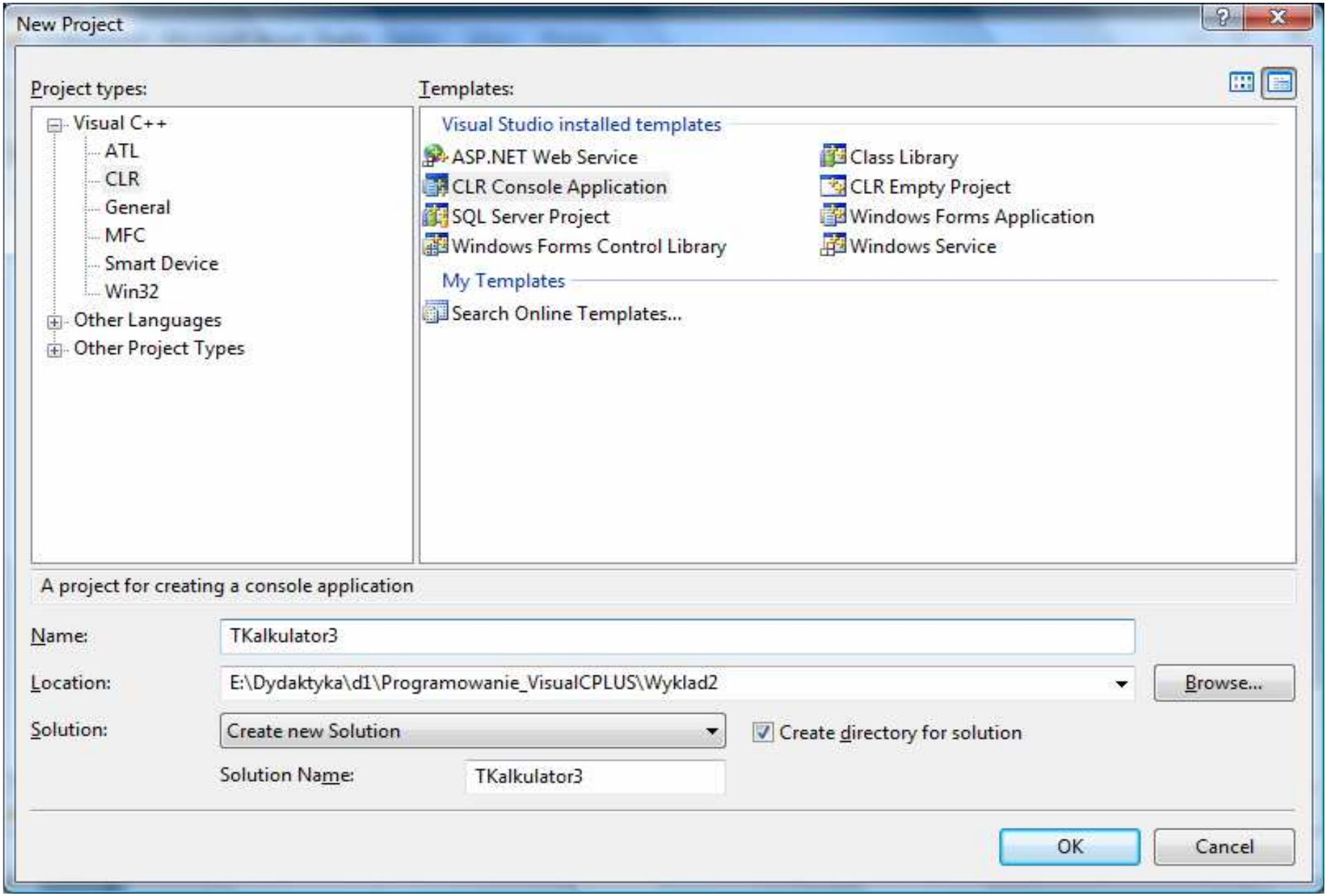
- **Tworzenie nowej listy metod obsluga_zdarzen1**

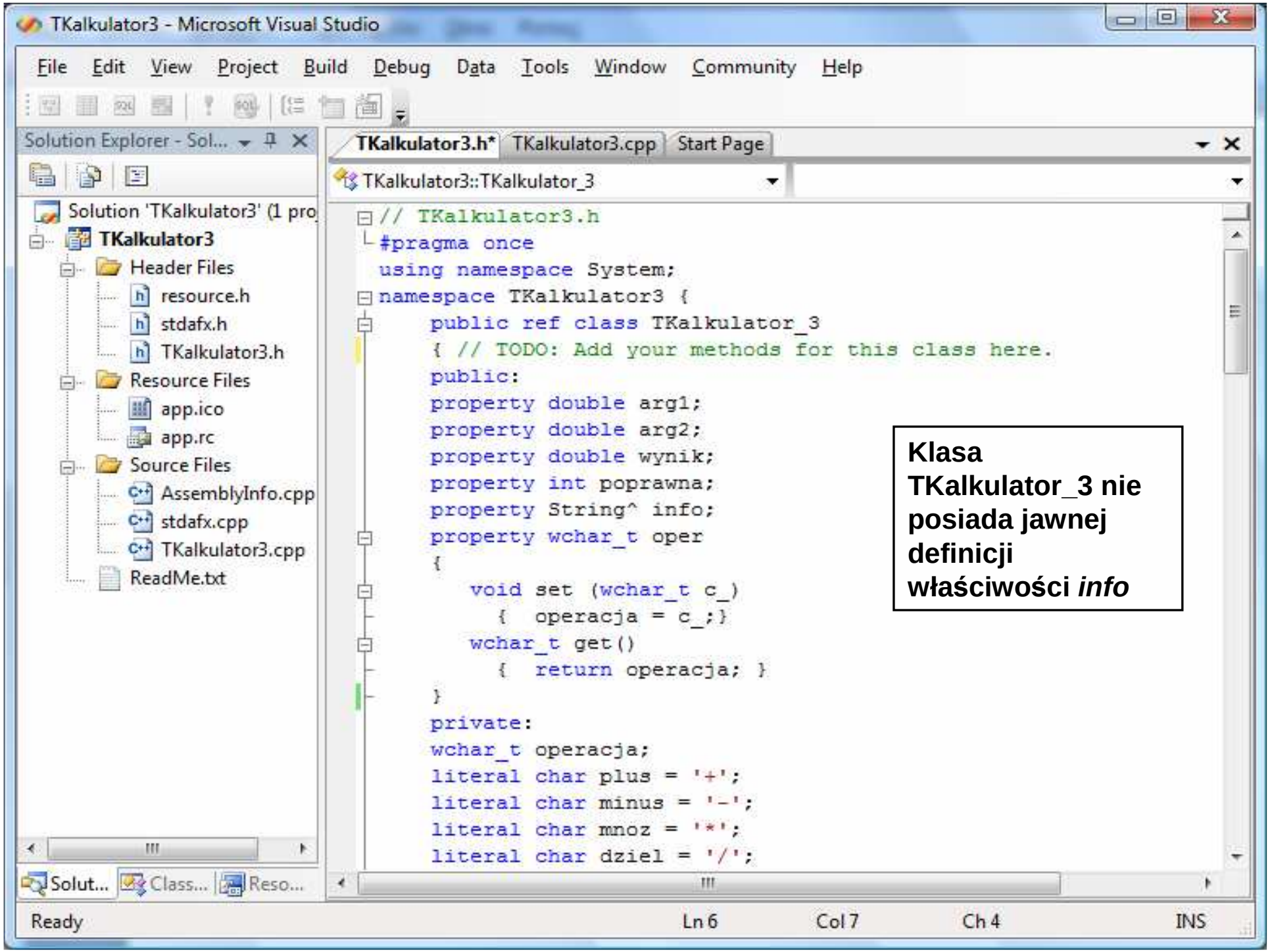
```
lista_Operacji^ lista_operacji1 =
  gcnew lista_Operacji (kalkulator,&Dzialania_kalkulatora::info2);
```

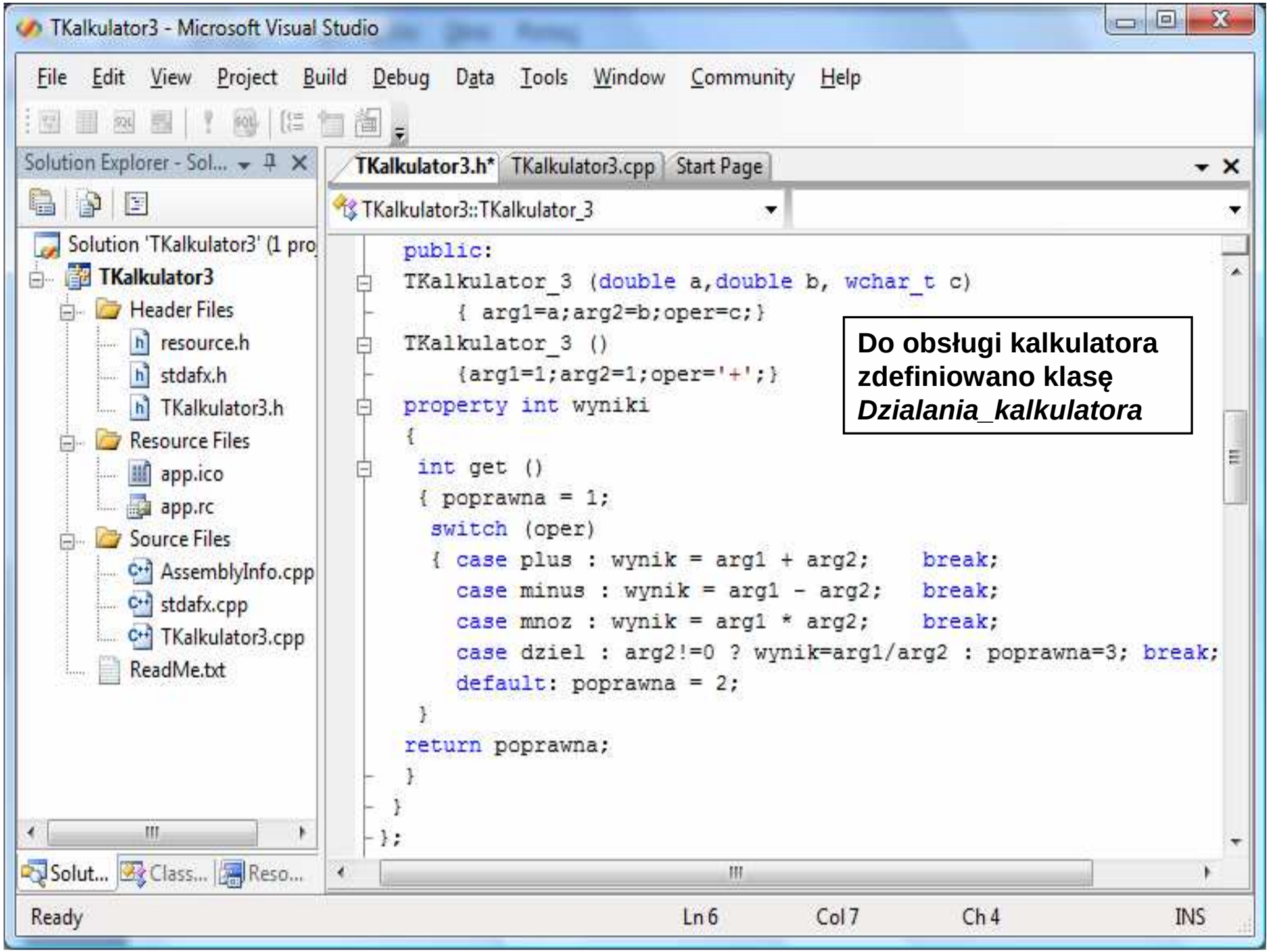
- **Utworzenie sumy list**

```
lista_operacji += lista_operacji1;
```

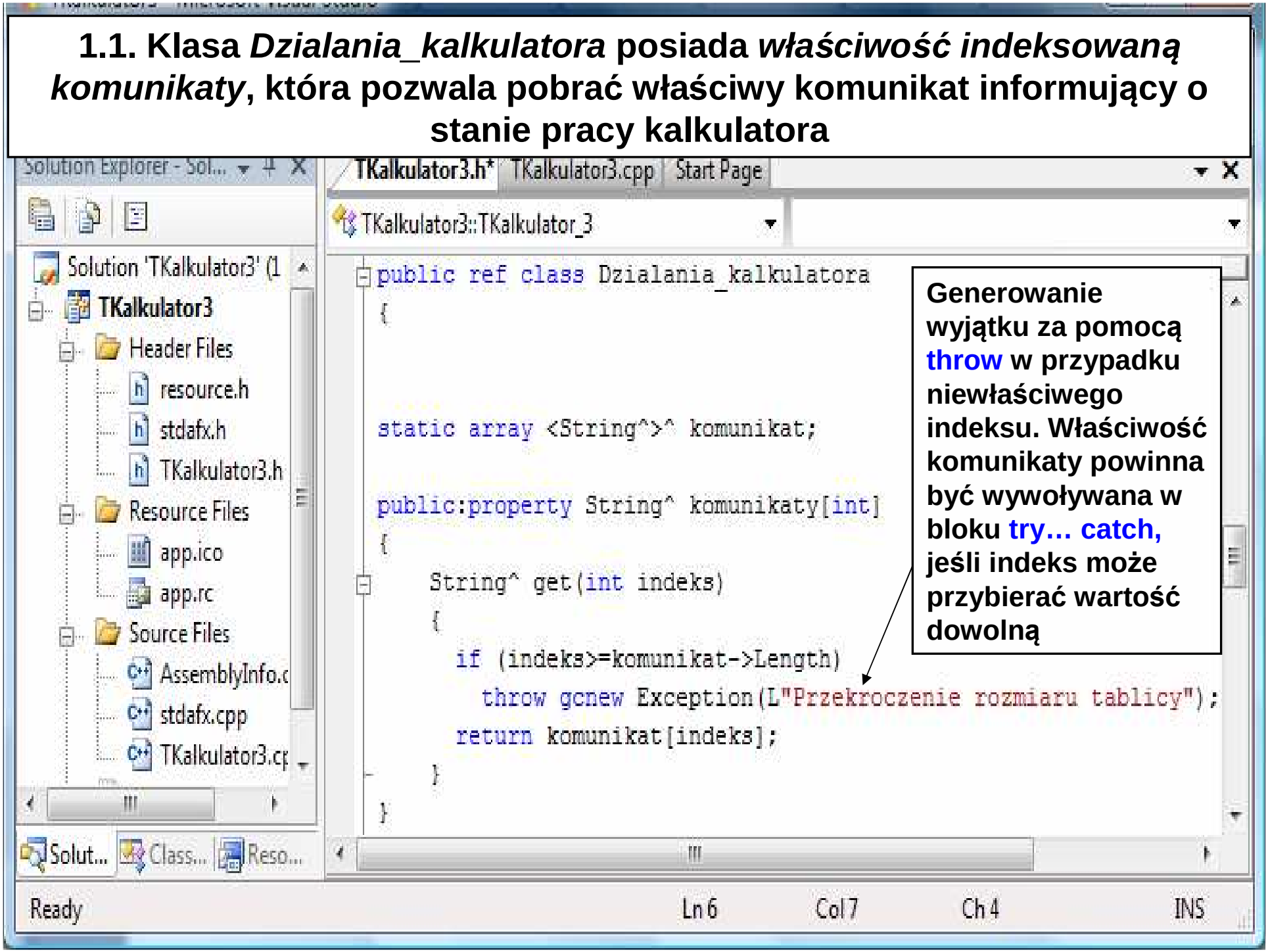
Zakładanie projektu CLR Console Application z plikiem TKalkulator3

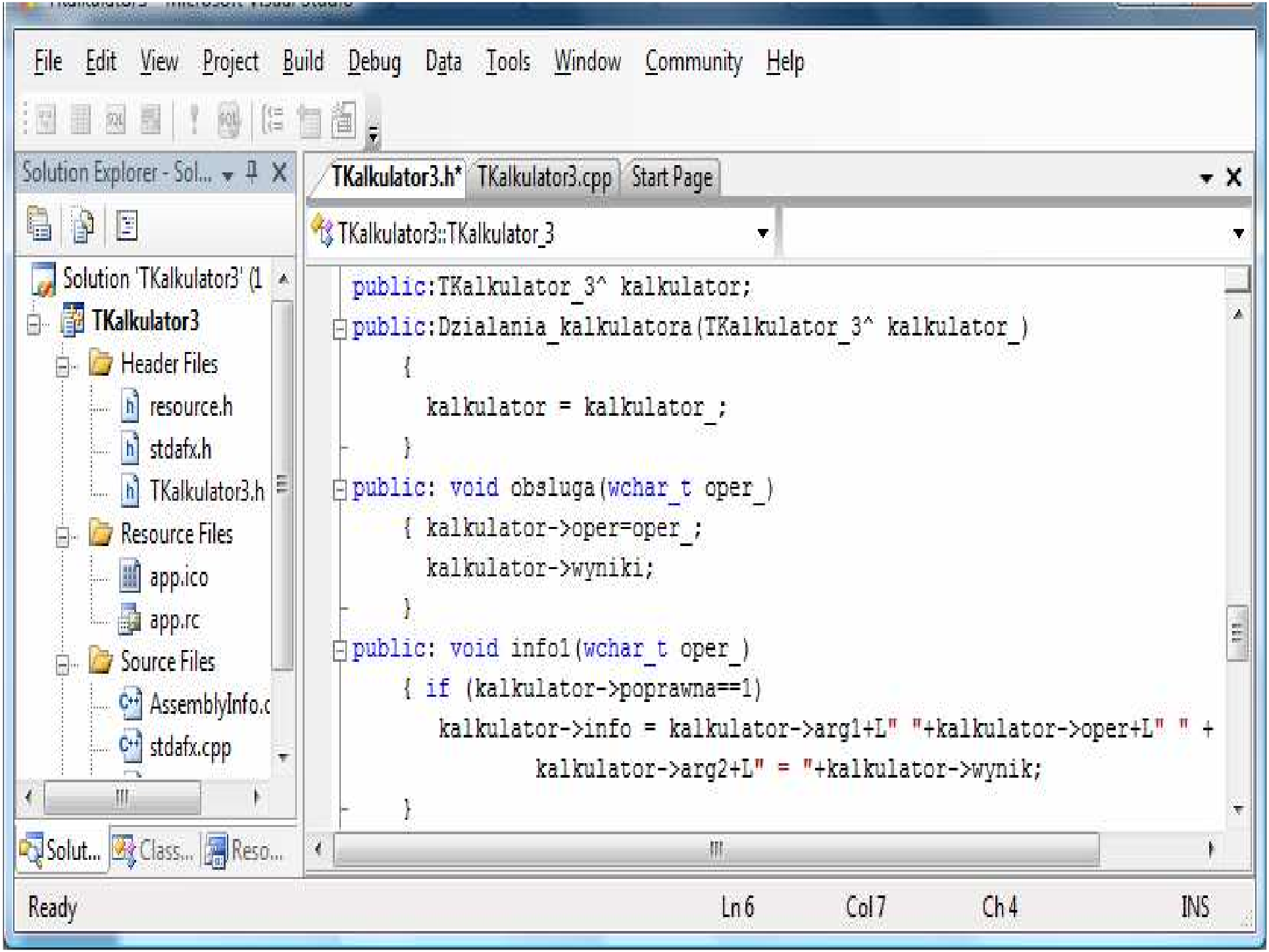


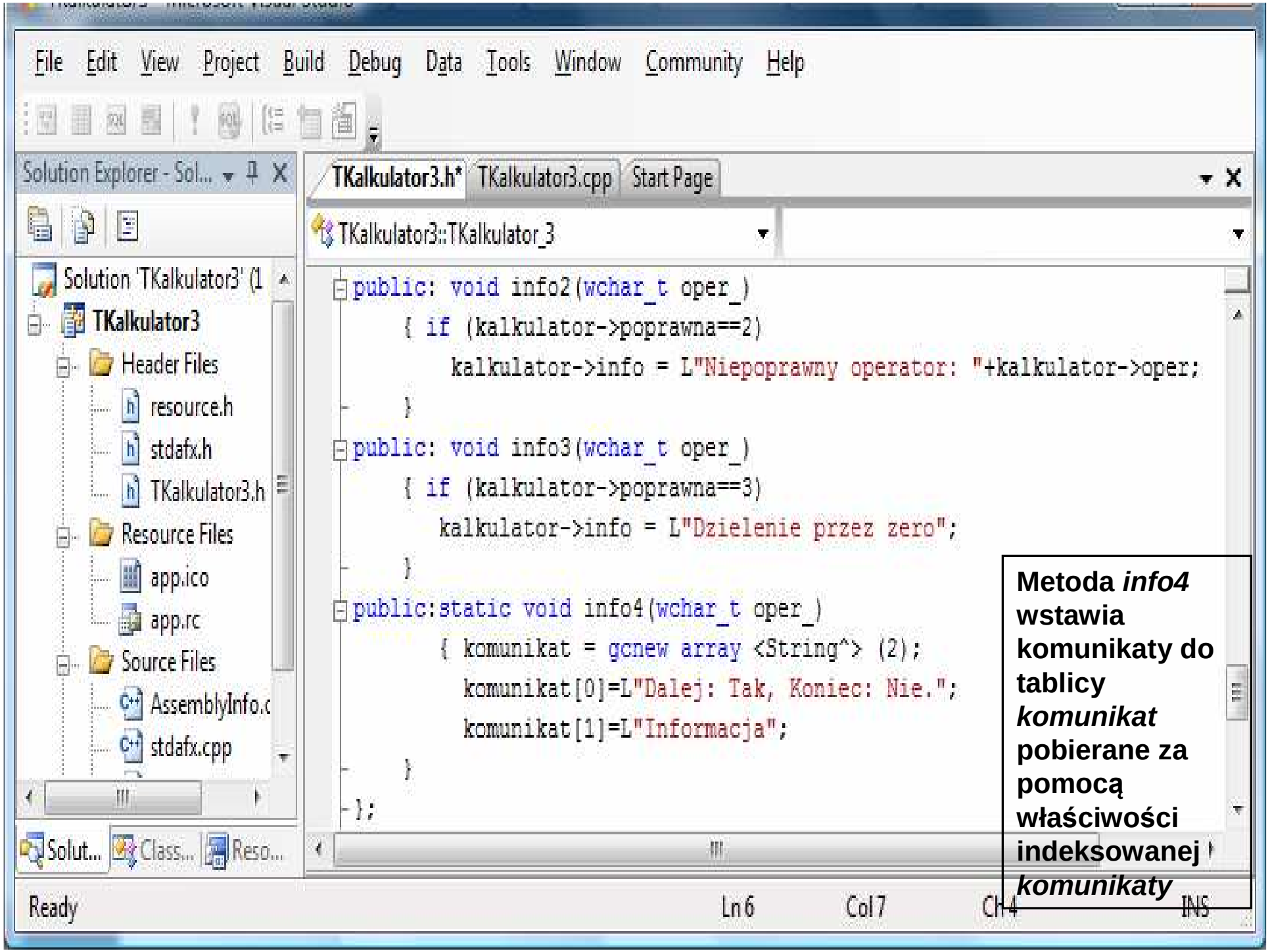

TKalkulator3 - Microsoft Visual Studio
File Edit View Project Build Debug Data Tools Window Community Help
Solution Explorer - Sol...
Solution 'TKalkulator3' (1 pro
TKalkulator3
Header Files
resource.h
stdafx.h
TKalkulator3.h
Resource Files
app.ico
app.rc
Source Files
AssemblyInfo.cpp
stdafx.cpp
TKalkulator3.cpp
ReadMe.txt
Solut...
Class...
Reso...
TKalkulator3.h*
TKalkulator3.cpp
Start Page
TKalkulator3::TKalkulator_3
// TKalkulator3.h
#pragma once
using namespace System;
namespace TKalkulator3 {
public ref class TKalkulator_3
{ // TODO: Add your methods for this class here.
public:
property double arg1;
property double arg2;
property double wynik;
property int poprawna;
property String^ info;
property wchar_t oper
{
void set (wchar_t c_)
{ operacja = c_;}
wchar_t get()
{ return operacja; }
}
private:
wchar_t operacja;
literal char plus = '+';
literal char minus = '-';
literal char mnoz = '*';
literal char dziel = '/';
Klasa TKalkulator_3 nie posiada jawnej definicji właściwości info
Ready
Ln 6
Col 7
Ch 4
INS


TKalkulator3 - Microsoft Visual Studio
File Edit View Project Build Debug Data Tools Window Community Help
Solution Explorer - Sol...
Solution 'TKalkulator3' (1 pro
TKalkulator3
Header Files
resource.h
stdafx.h
TKalkulator3.h
Resource Files
app.ico
app.rc
Source Files
AssemblyInfo.cpp
stdafx.cpp
TKalkulator3.cpp
ReadMe.txt
Solut...
Class...
Reso...
TKalkulator3.h*
TKalkulator3.cpp
Start Page
TKalkulator3::TKalkulator_3
public:
TKalkulator_3 (double a,double b, wchar_t c)
{ arg1=a;arg2=b;oper=c;}
TKalkulator_3 ()
{arg1=1;arg2=1;oper='+';}
property int wyniki
{
int get ()
{ poprawna = 1;
switch (oper)
{ case plus : wynik = arg1 + arg2; break;
case minus : wynik = arg1 - arg2; break;
case mnoz : wynik = arg1 * arg2; break;
case dziel : arg2!=0 ? wynik=arg1/arg2 : poprawna=3; break;
default: poprawna = 2;
}
return poprawna;
}
}
};
Do obsługi kalkulatora zdefiniowano klasę Dzialania_kalkulatora
Ready
Ln 6
Col 7
Ch 4
INS

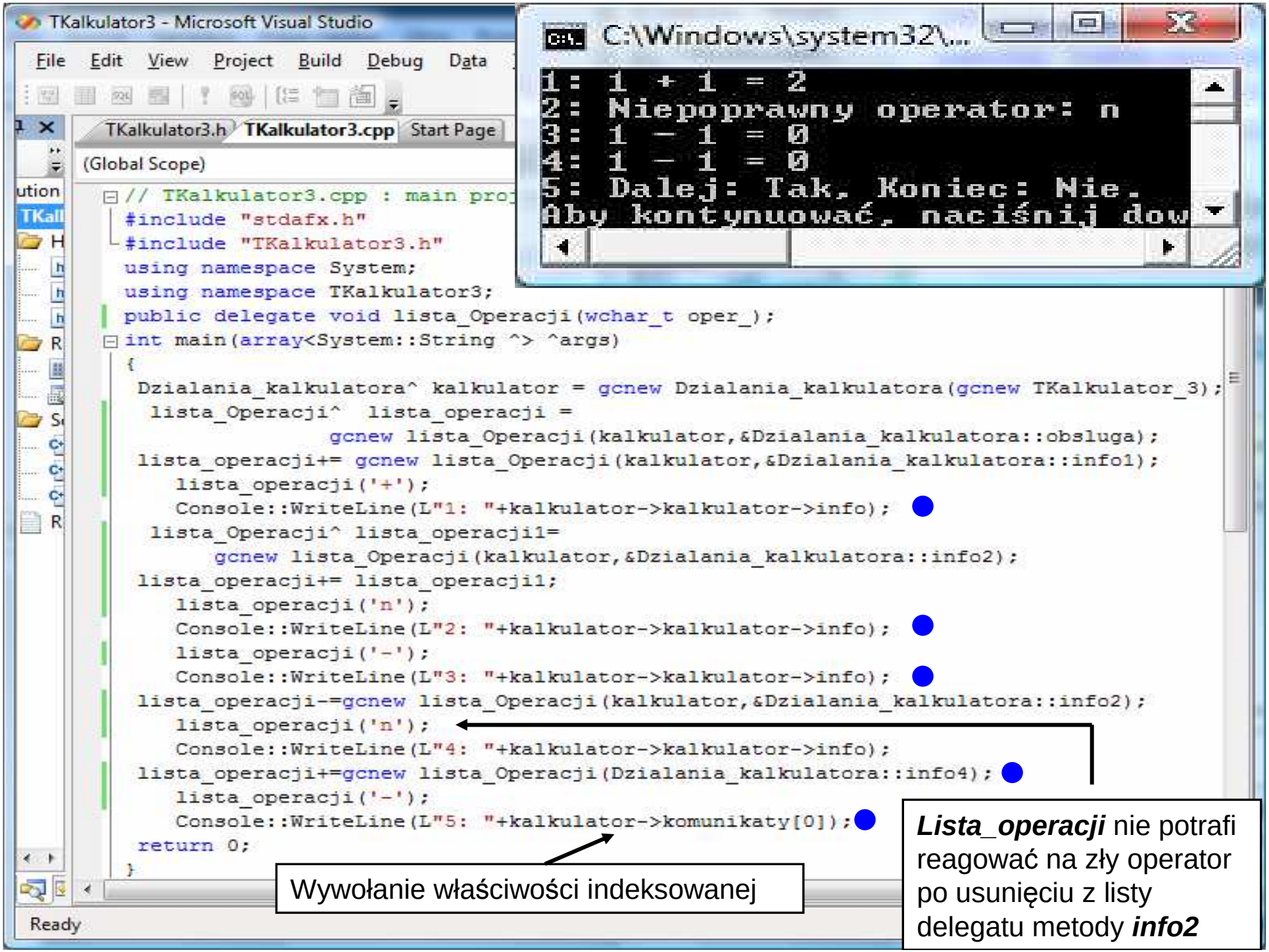
1.1. Klasa *Dzialania_kalkulatora* posiada *właściwość indeksowaną komunikaty*, która pozwala pobrać właściwy komunikat informujący o stanie pracy kalkulatora




File Edit View Project Build Debug Data Tools Window Community Help
Solution Explorer - Sol...
Solution 'TKalkulator3' (1
TKalkulator3
Header Files
resource.h
stdafx.h
TKalkulator3.h
Resource Files
app.ico
app.rc
Source Files
AssemblyInfo.c
stdafx.cpp
Solut...
Class...
Reso...
TKalkulator3.h*
TKalkulator3.cpp
Start Page
TKalkulator3::TKalkulator_3
public:TKalkulator_3^ kalkulator;
public:Dzialania_kalkulatora(TKalkulator_3^ kalkulator_)
{
kalkulator = kalkulator_;
}
public: void obsluga(wchar_t oper_)
{ kalkulator->oper=oper_;
kalkulator->wyniki;
}
public: void info1(wchar_t oper_)
{ if (kalkulator->poprawna==1)
kalkulator->info = kalkulator->arg1+L" "+kalkulator->oper+L" " +
kalkulator->arg2+L" = "+kalkulator->wynik;
}
Ready
Ln 6
Col 7
Ch 4
INS



Metoda *info4* wstawia komunikaty do tablicy *komunikat* pobierane za pomocą właściwości indeksowanej *komunikaty*


TKalkulator3 - Microsoft Visual Studio
File Edit View Project Build Debug Data
TKalkulator3.h TKalkulator3.cpp Start Page
(Global Scope)
// TKalkulator3.cpp : main proj
#include "stdafx.h"
#include "TKalkulator3.h"
using namespace System;
using namespace TKalkulator3;
public delegate void lista_Operacji(wchar_t oper_);
int main(array<System::String ^> ^args)
{
Dzialania_kalkulatora^ kalkulator = gcnew Dzialania_kalkulatora(gcnew TKalkulator_3);
lista_Operacji^ lista_operacji =
gcnew lista_Operacji(kalkulator,&Dzialania_kalkulatora::obsluga);
lista_operacji+= gcnew lista_Operacji(kalkulator,&Dzialania_kalkulatora::info1);
lista_operacji('+');
Console::WriteLine(L"1: "+kalkulator->kalkulator->info);
lista_Operacji^ lista_operacji1=
gcnew lista_Operacji(kalkulator,&Dzialania_kalkulatora::info2);
lista_operacji+= lista_operacji1;
lista_operacji('n');
Console::WriteLine(L"2: "+kalkulator->kalkulator->info);
lista_operacji('-');
Console::WriteLine(L"3: "+kalkulator->kalkulator->info);
lista_operacji-=gcnew lista_Operacji(kalkulator,&Dzialania_kalkulatora::info2);
lista_operacji('n');
Console::WriteLine(L"4: "+kalkulator->kalkulator->info);
lista_operacji+=gcnew lista_Operacji(Dzialania_kalkulatora::info4);
lista_operacji('-');
Console::WriteLine(L"5: "+kalkulator->komunikaty[0]);
return 0;
}
C:\Windows\system32\...
1: 1 + 1 = 2
2: Niepoprawny operator: n
3: 1 - 1 = 0
4: 1 - 1 = 0
5: Dalej: Tak, Koniec: Nie.
Aby kontynuować, naciśnij dow
Wywołanie właściwości indeksowanej
Lista_operacji nie potrafi reagować na zły operator po usunięciu z listy delegatu metody info2
Ready

2. Delegaty niewiązane (2)

Definicja delegatu niewiązanego

Delegat niewiązany wskazuje na funkcje obiektowe o określonej liście parametrów dla danego typu obiektu i może być wywołany dla wielu obiektów typu użytego w definicji.

```
public delegate void lista_Operacji1(Dzialania_kalkulatora^,wchar_t c);
```

- dodano do listy metod klasy *Dzialania_kalkulatora* metodę *obsluga*

```
lista_Operacji1^  lista_operacji =
        gcnew lista_Operacji1(&Dzialania_kalkulatora::obsluga);
```

- na obiekcie *kalkulator1* wykonuje się metodę *obsluga* dla operacji odejmowanie

```
lista_operacji(kalkulator1,'-');
```

- dodano do listy metod klasy *Dzialania_kalkulatora* metodę *info1*

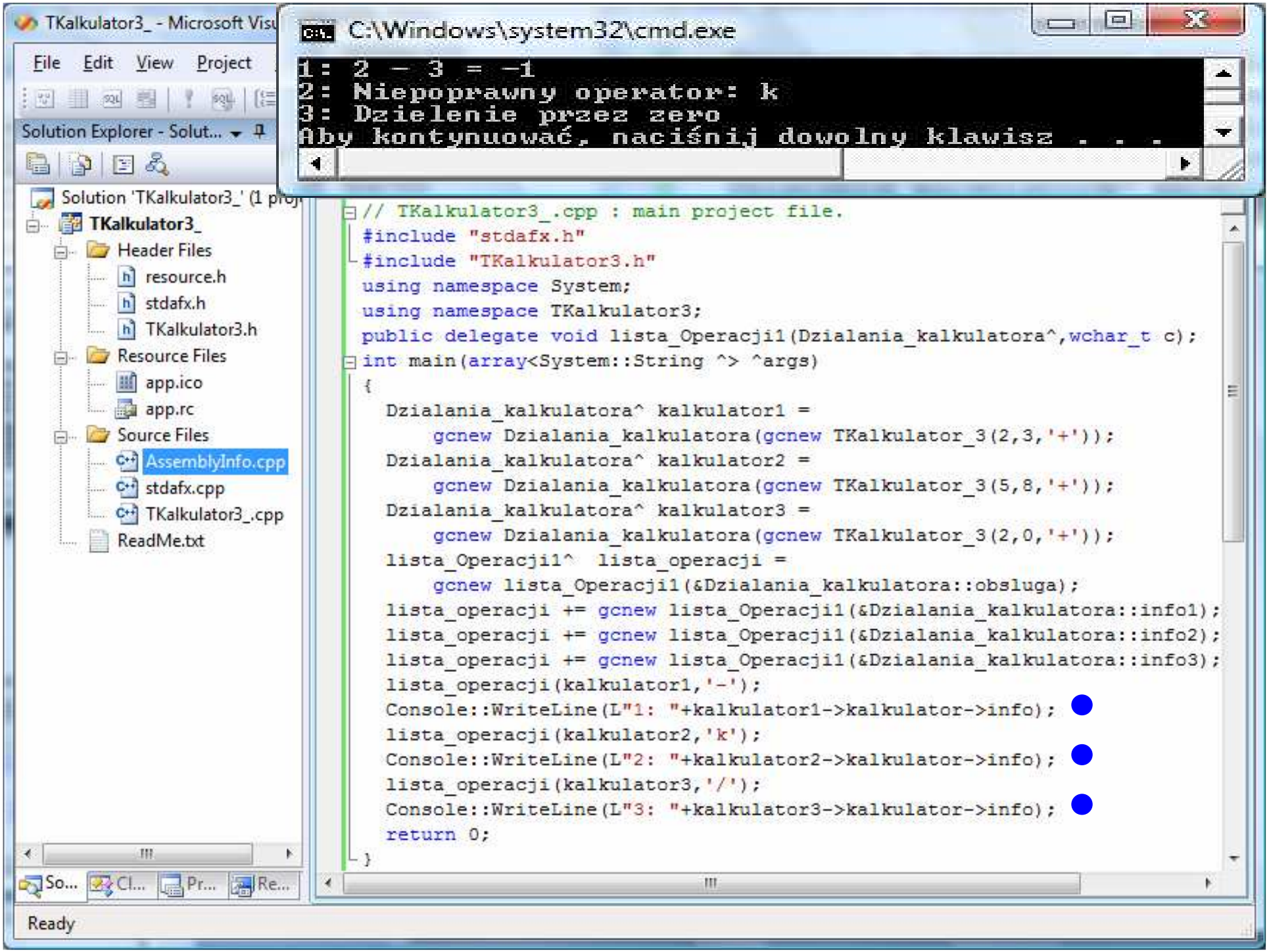
```
lista_operacji += gcnew lista_Operacji1(&Dzialania_kalkulatora::info1);
```

- na obiekcie *kalkulator2* wykonuje się metody *obsluga* dla operacji mnożenie oraz *info1*

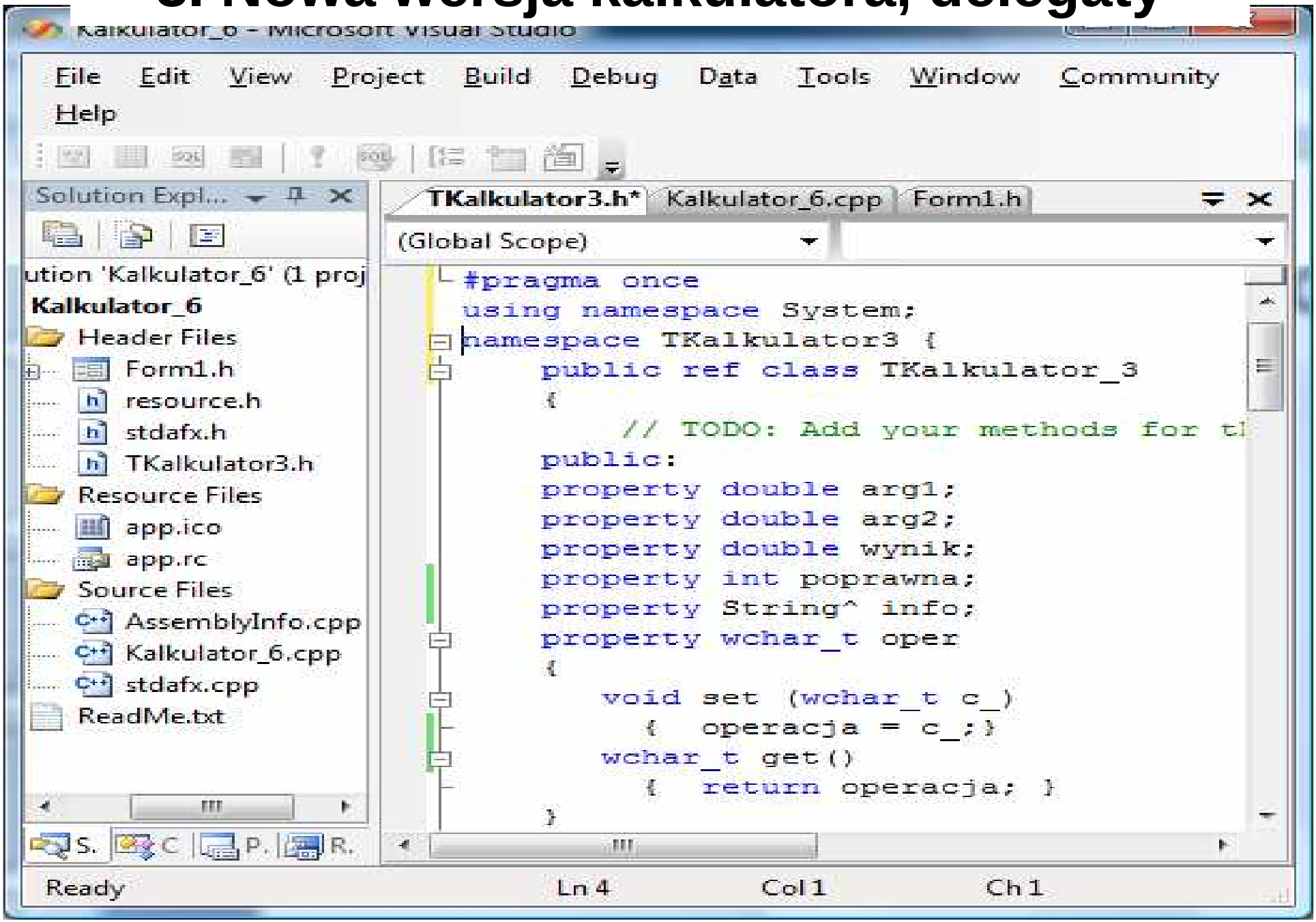
```
lista_operacji(kalkulator2,‘*');
```

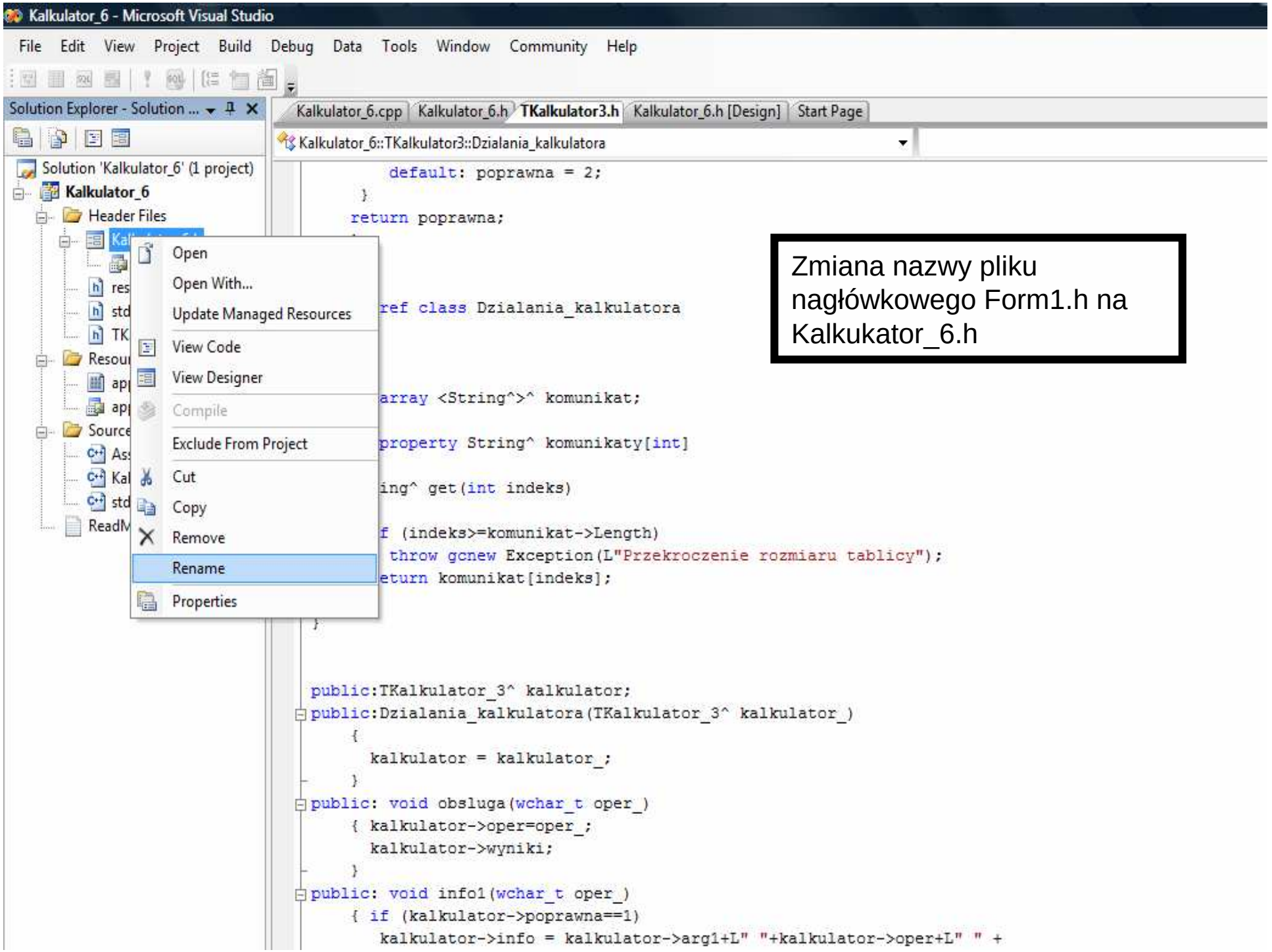
Delegaty niewiązane (2)

```
// TKalkulator3_.cpp : main project file.
#include "stdafx.h"
#include "TKalkulator3.h"
using namespace System;
using namespace TKalkulator3;
public delegate void lista_Operacji1(Dzialania_kalkulatora^,wchar_t c);
int main(array<System::String ^> ^args)
{ Dzialania_kalkulatora^ kalkulator1 =
          gcnew Dzialania_kalkulatora(gcnew TKalkulator_3(2,3,'+'));
  Dzialania_kalkulatora^ kalkulator2 =
          gcnew Dzialania_kalkulatora(gcnew TKalkulator_3(5,8,'+'));
  Dzialania_kalkulatora^ kalkulator3 =
         gcnew Dzialania_kalkulatora(gcnew TKalkulator_3(2,0,'+'));
  lista_Operacji1^  lista_operacji =
         gcnew lista_Operacji1(&Dzialania_kalkulatora::obsluga);
  lista_operacji += gcnew lista_Operacji1(&Dzialania_kalkulatora::info1);
  lista_operacji += gcnew lista_Operacji1(&Dzialania_kalkulatora::info2);
  lista_operacji += gcnew lista_Operacji1(&Dzialania_kalkulatora::info3);
  lista_operacji(kalkulator1,'-');   Console::WriteLine(L"1: "+kalkulator1->kalkulator->info);
  lista_operacji(kalkulator2,'k');  Console::WriteLine(L"2: "+kalkulator2->kalkulator->info);
  lista_operacji(kalkulator3,'/');   Console::WriteLine(L"3: "+kalkulator3->kalkulator->info);
  return 0;
}
```


TKalkulator3_ - Microsoft Visu
File Edit View Project
Solution Explorer - Solut...
Solution 'TKalkulator3_' (1 proj
TKalkulator3_
Header Files
resource.h
stdafx.h
TKalkulator3.h
Resource Files
app.ico
app.rc
Source Files
AssemblyInfo.cpp
stdafx.cpp
TKalkulator3_.cpp
ReadMe.txt
So...
Cl...
Pr...
Re...
C:\Windows\system32\cmd.exe
1: 2 - 3 = -1
2: Niepoprawny operator: k
3: Dzielenie przez zero
Aby kontynuować, naciśnij dowolny klawisz . . .
// TKalkulator3_.cpp : main project file.
#include "stdafx.h"
#include "TKalkulator3.h"
using namespace System;
using namespace TKalkulator3;
public delegate void lista_Operacji1(Dzialania_kalkulatora^,wchar_t c);
int main(array<System::String ^> ^args)
{
Dzialania_kalkulatora^ kalkulator1 =
gcnew Dzialania_kalkulatora(gcnew TKalkulator_3(2,3,'+'));
Dzialania_kalkulatora^ kalkulator2 =
gcnew Dzialania_kalkulatora(gcnew TKalkulator_3(5,8,'+'));
Dzialania_kalkulatora^ kalkulator3 =
gcnew Dzialania_kalkulatora(gcnew TKalkulator_3(2,0,'+'));
lista_Operacji1^ lista_operacji =
gcnew lista_Operacji1(&Dzialania_kalkulatora::obsluga);
lista_operacji += gcnew lista_Operacji1(&Dzialania_kalkulatora::info1);
lista_operacji += gcnew lista_Operacji1(&Dzialania_kalkulatora::info2);
lista_operacji += gcnew lista_Operacji1(&Dzialania_kalkulatora::info3);
lista_operacji(kalkulator1,'-');
Console::WriteLine(L"1: "+kalkulator1->kalkulator->info);
lista_operacji(kalkulator2,'k');
Console::WriteLine(L"2: "+kalkulator2->kalkulator->info);
lista_operacji(kalkulator3,'/');
Console::WriteLine(L"3: "+kalkulator3->kalkulator->info);
return 0;
}
Ready

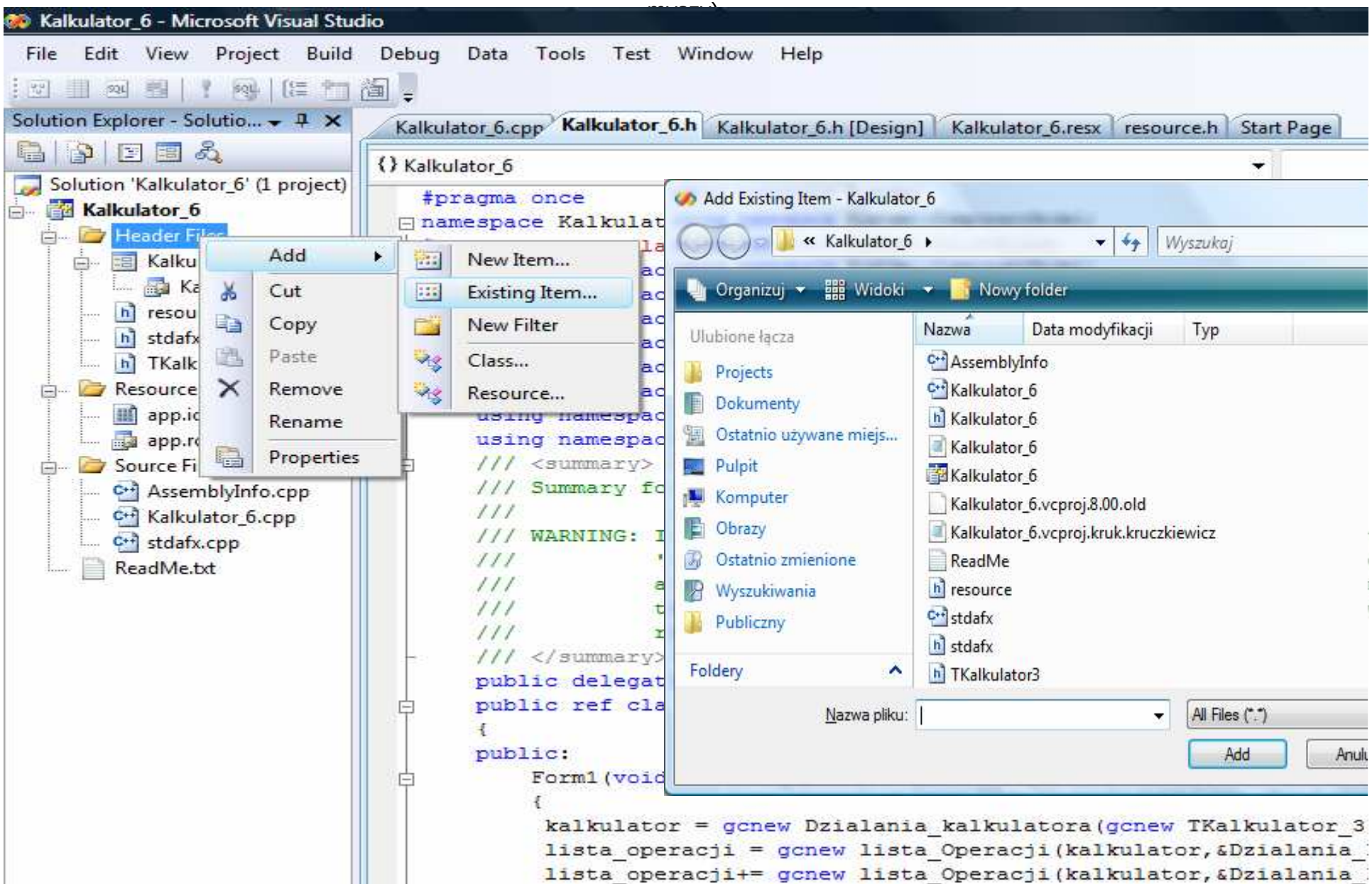
3. Nowa wersja kalkulatora, delegaty



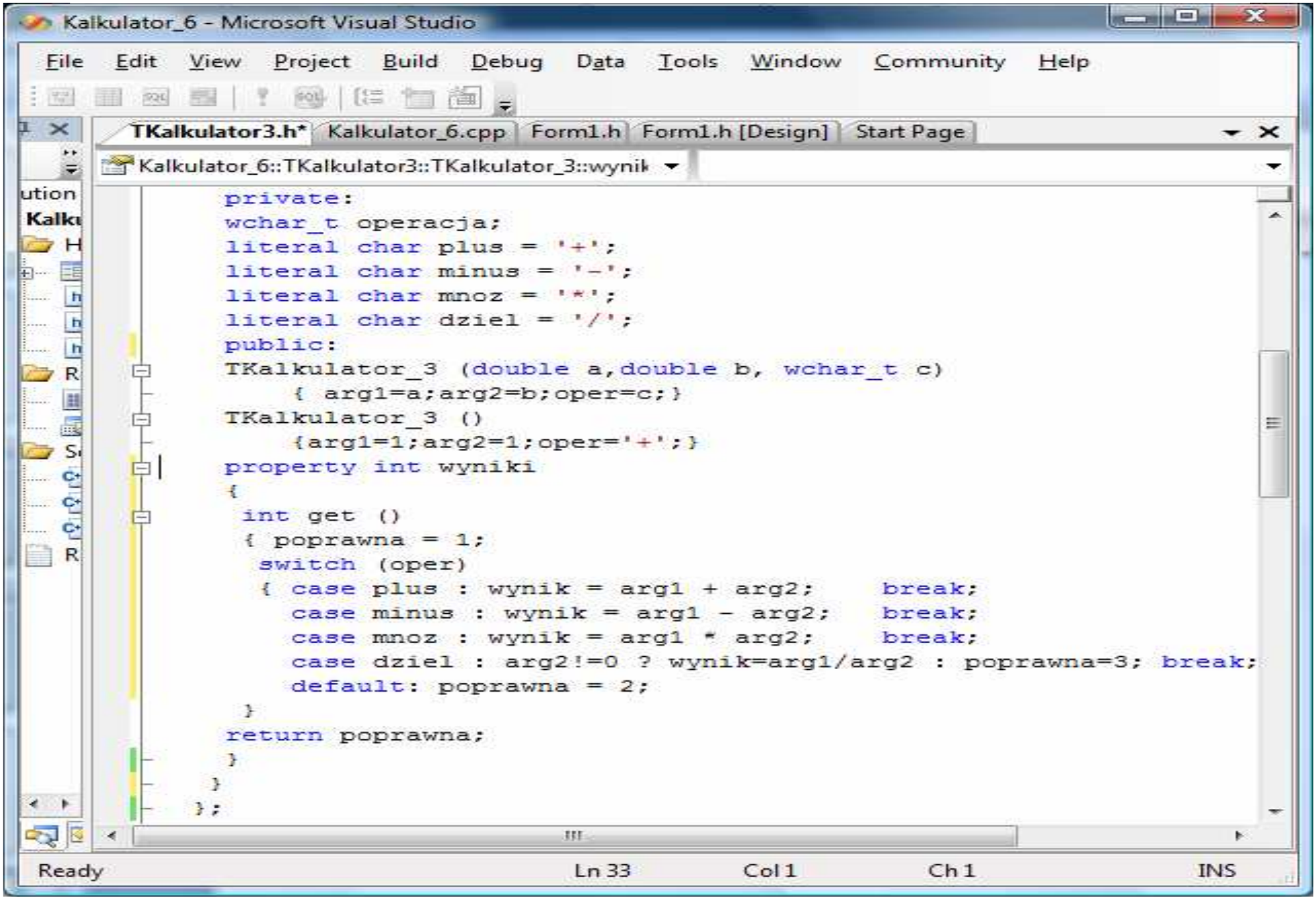


Zmiana nazwy pliku nagłówkowego Form1.h na Kalkukator_6.h

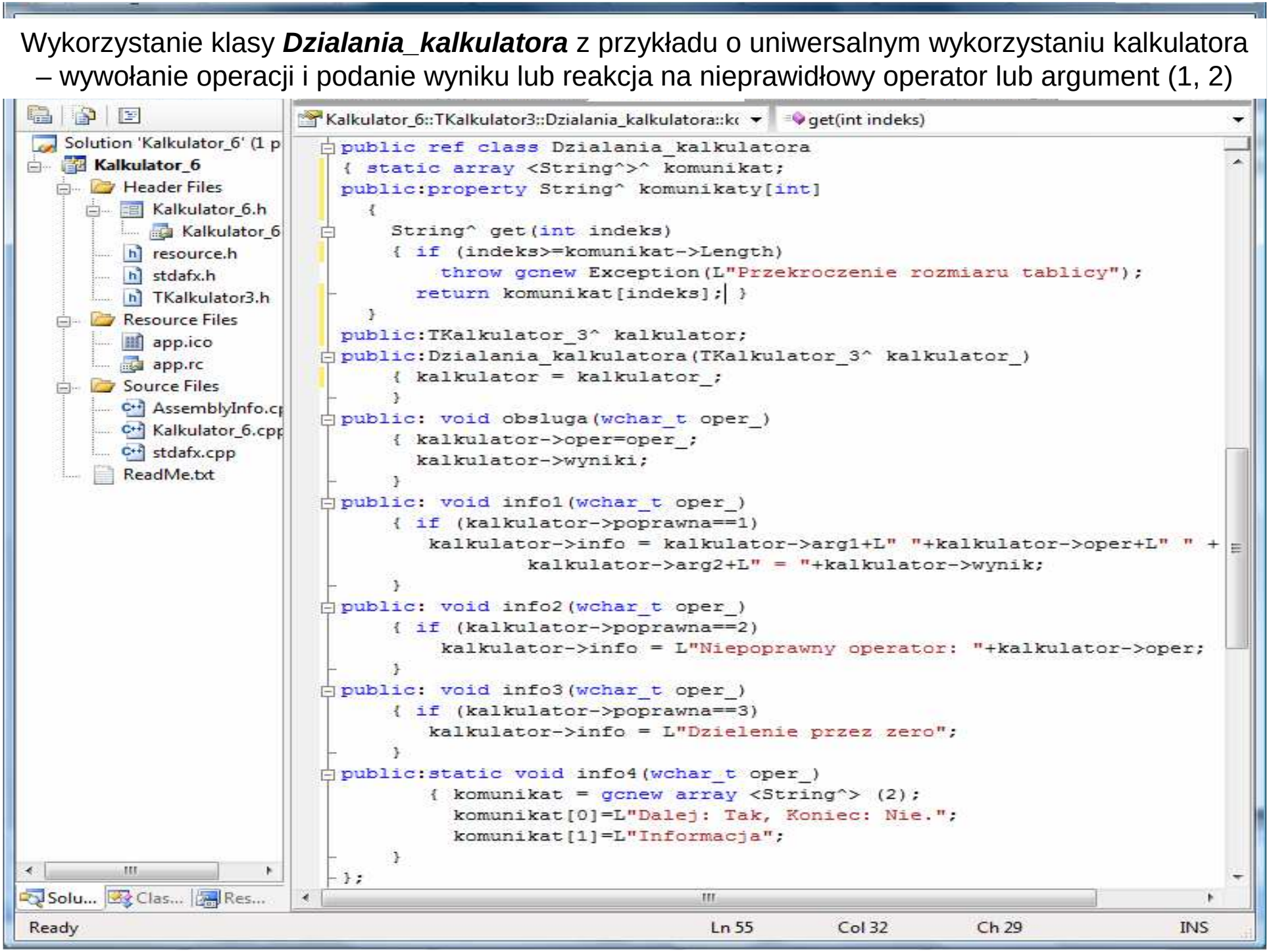
Dołączenie do projektu pliku nagłówkowego TKalkulator3.h z projektu konsolowego z p.2. (skopiowano plik TKalkulator3.h do katalogu Kalkulator_6/Kalkulator_6 z katalogu TKalkulator3/TKalkulator3 i dodano go do projektu Kalkulator_6 za pomocą wyskakującego menu wywołanego prawym klawiszem



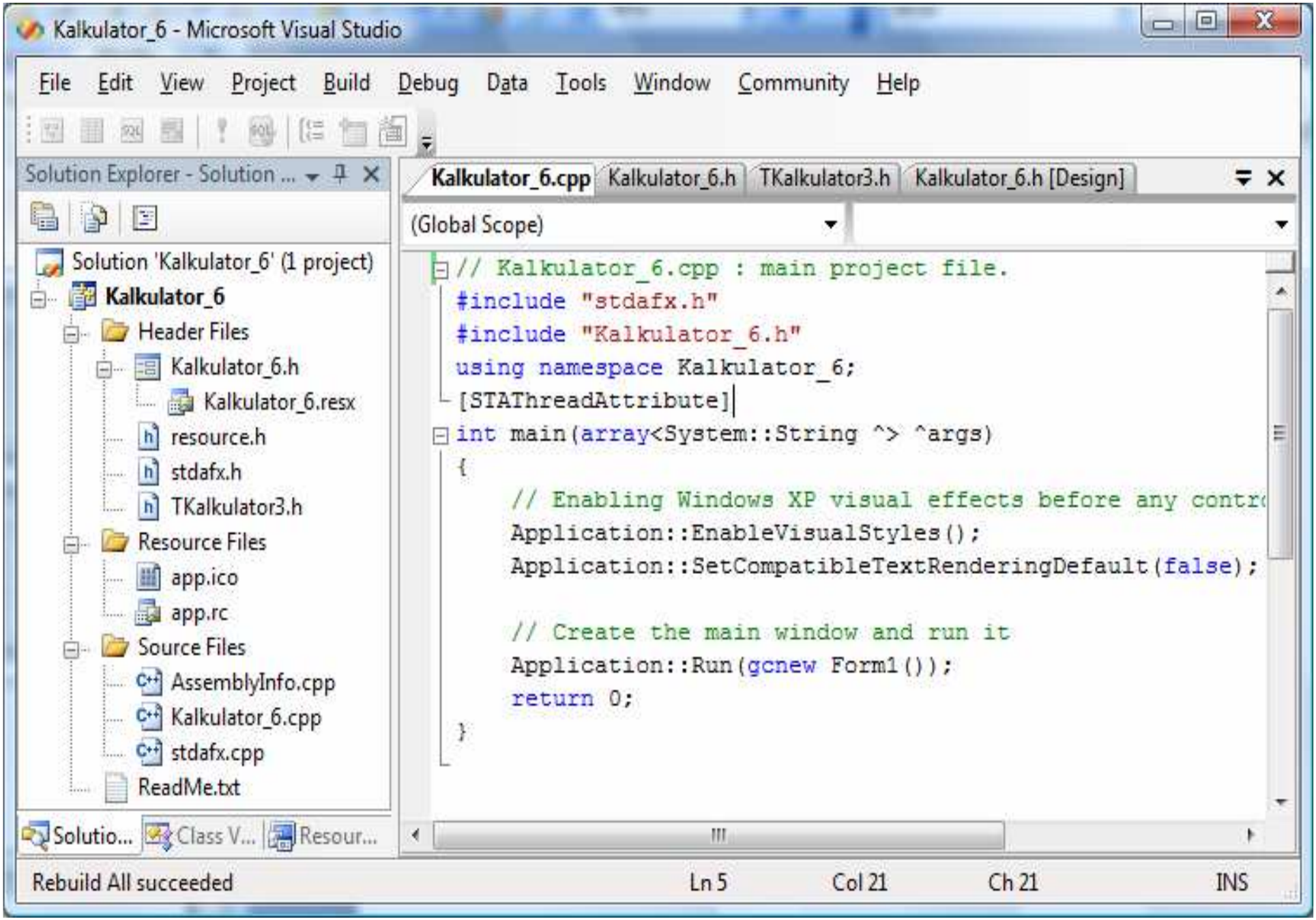
Zawartość pliku TKalkulator3.h

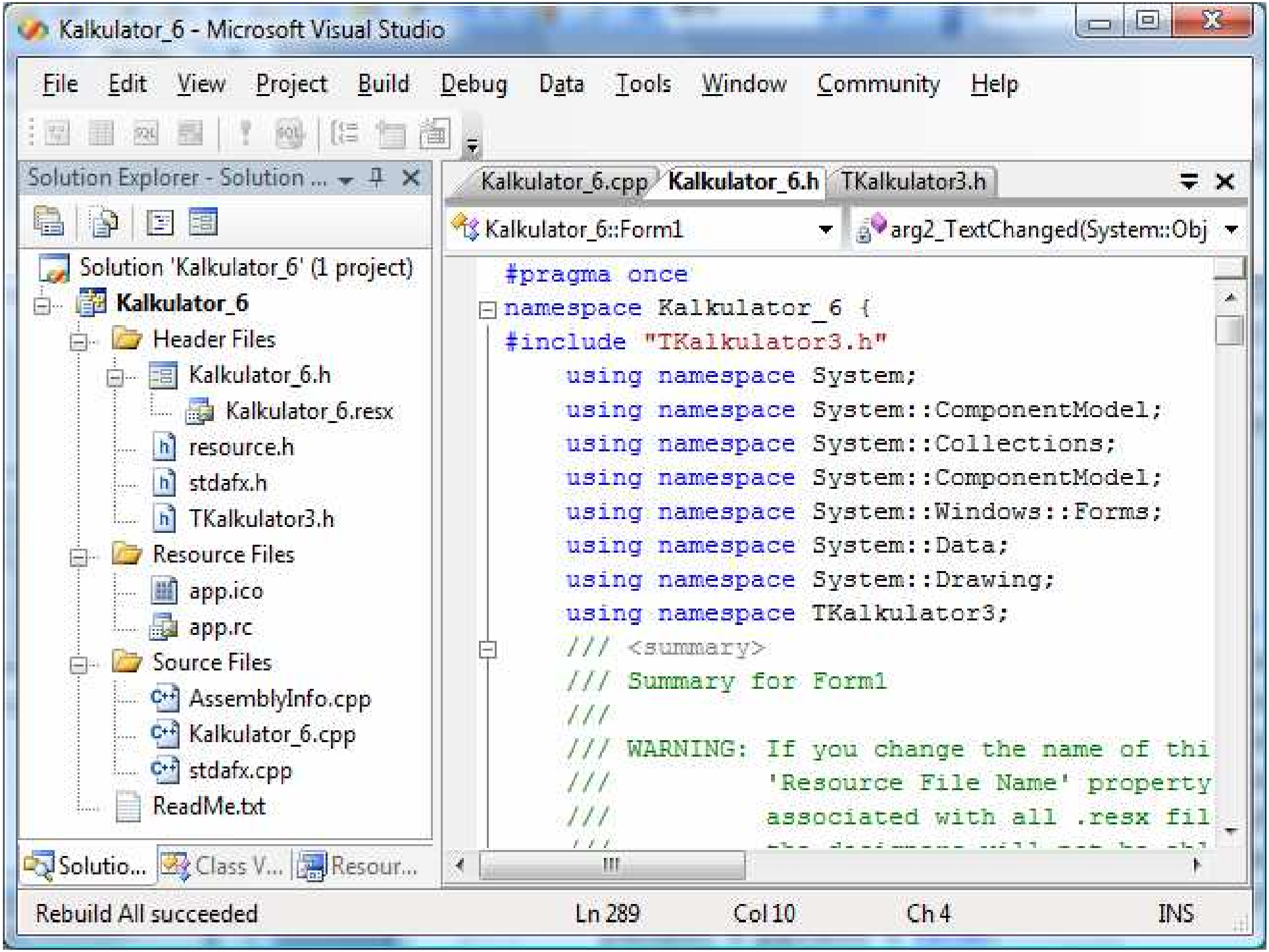


Wykorzystanie klasy ***Dzialania_kalkulatora*** z przykładu o uniwersalnym wykorzystaniu kalkulatora – wywołanie operacji i podanie wyniku lub reakcja na nieprawidłowy operator lub argument (1, 2)

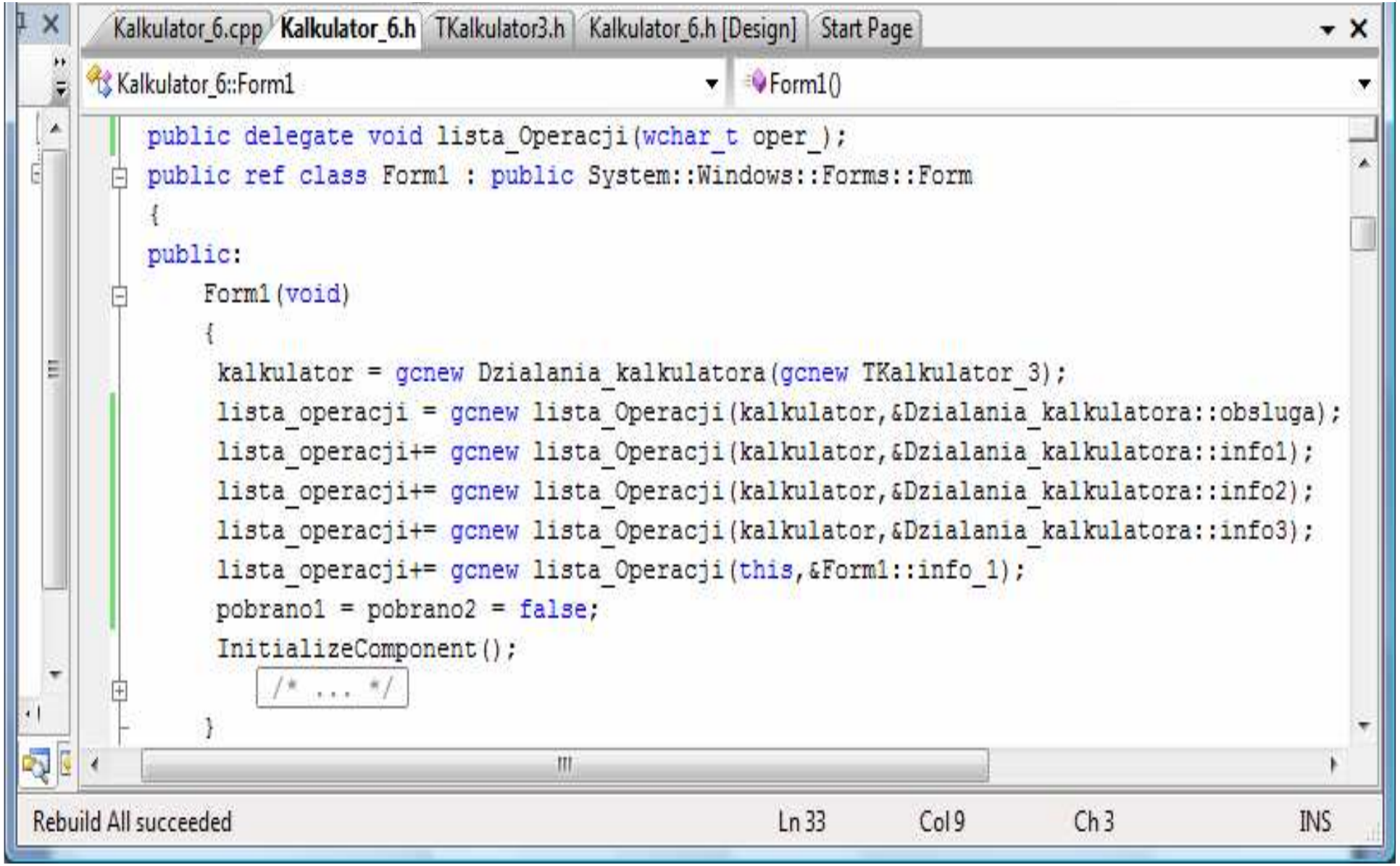


Zastosowanie klasy z uniwersalnym wykorzystaniem kalkulatora z GUI

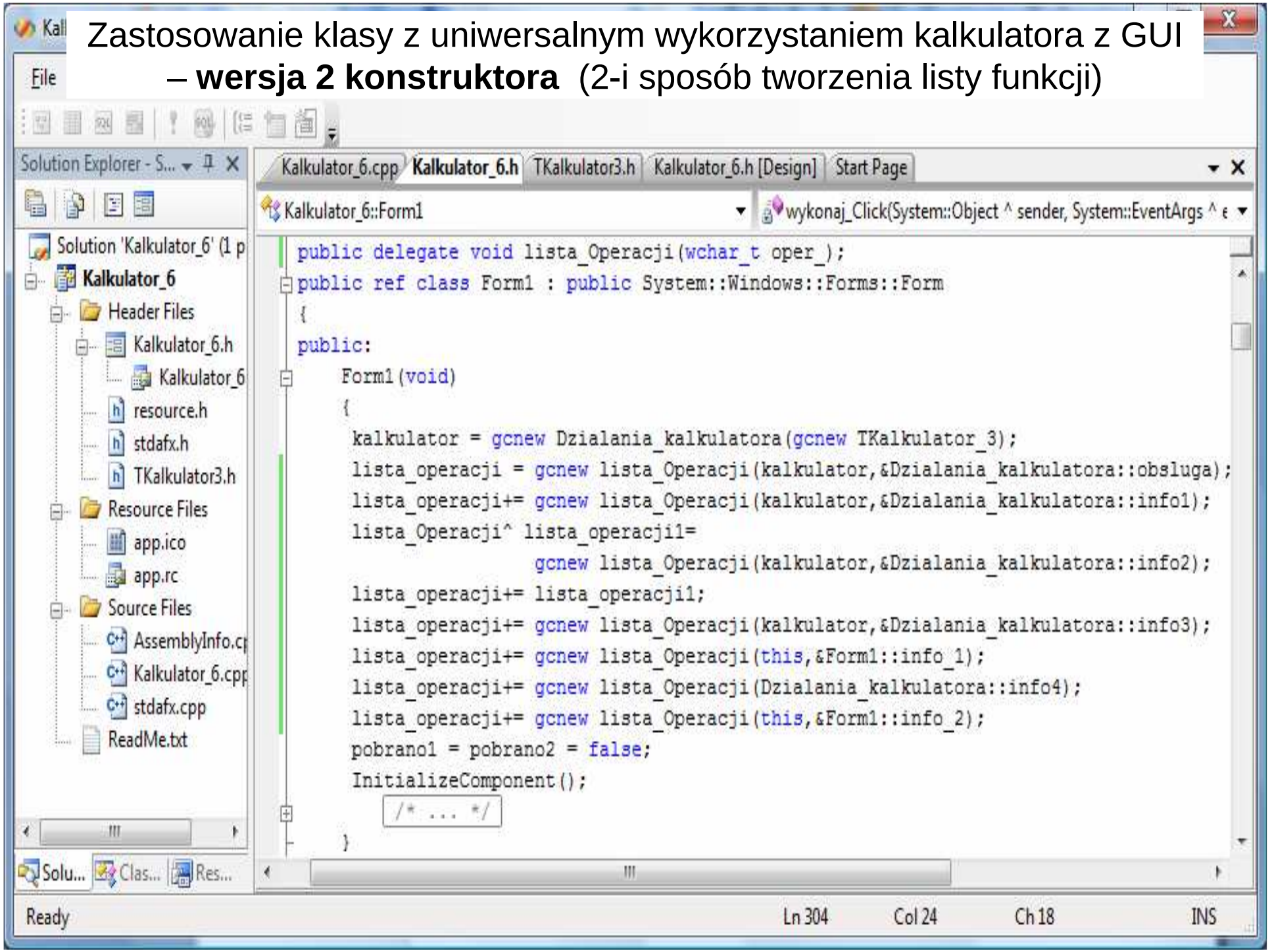



Kalkulator_6 - Microsoft Visual Studio
File Edit View Project Build Debug Data Tools Window Community Help
Solution Explorer - Solution ...
Solution 'Kalkulator_6' (1 project)
Kalkulator_6
Header Files
Kalkulator_6.h
Kalkulator_6.resx
resource.h
stdafx.h
TKalkulator3.h
Resource Files
app.ico
app.rc
Source Files
AssemblyInfo.cpp
Kalkulator_6.cpp
stdafx.cpp
ReadMe.txt
Solutio...
Class V...
Resour...
Kalkulator_6.cpp
Kalkulator_6.h
TKalkulator3.h
Kalkulator_6::Form1
arg2_TextChanged(System::Obj
#pragma once
namespace Kalkulator_6 {
#include "TKalkulator3.h"
using namespace System;
using namespace System::ComponentModel;
using namespace System::Collections;
using namespace System::ComponentModel;
using namespace System::Windows::Forms;
using namespace System::Data;
using namespace System::Drawing;
using namespace TKalkulator3;
/// <summary>
/// Summary for Form1
///
/// WARNING: If you change the name of thi
/// 'Resource File Name' property
/// associated with all .resx fil
Rebuild All succeeded
Ln 289
Col 10
Ch 4
INS

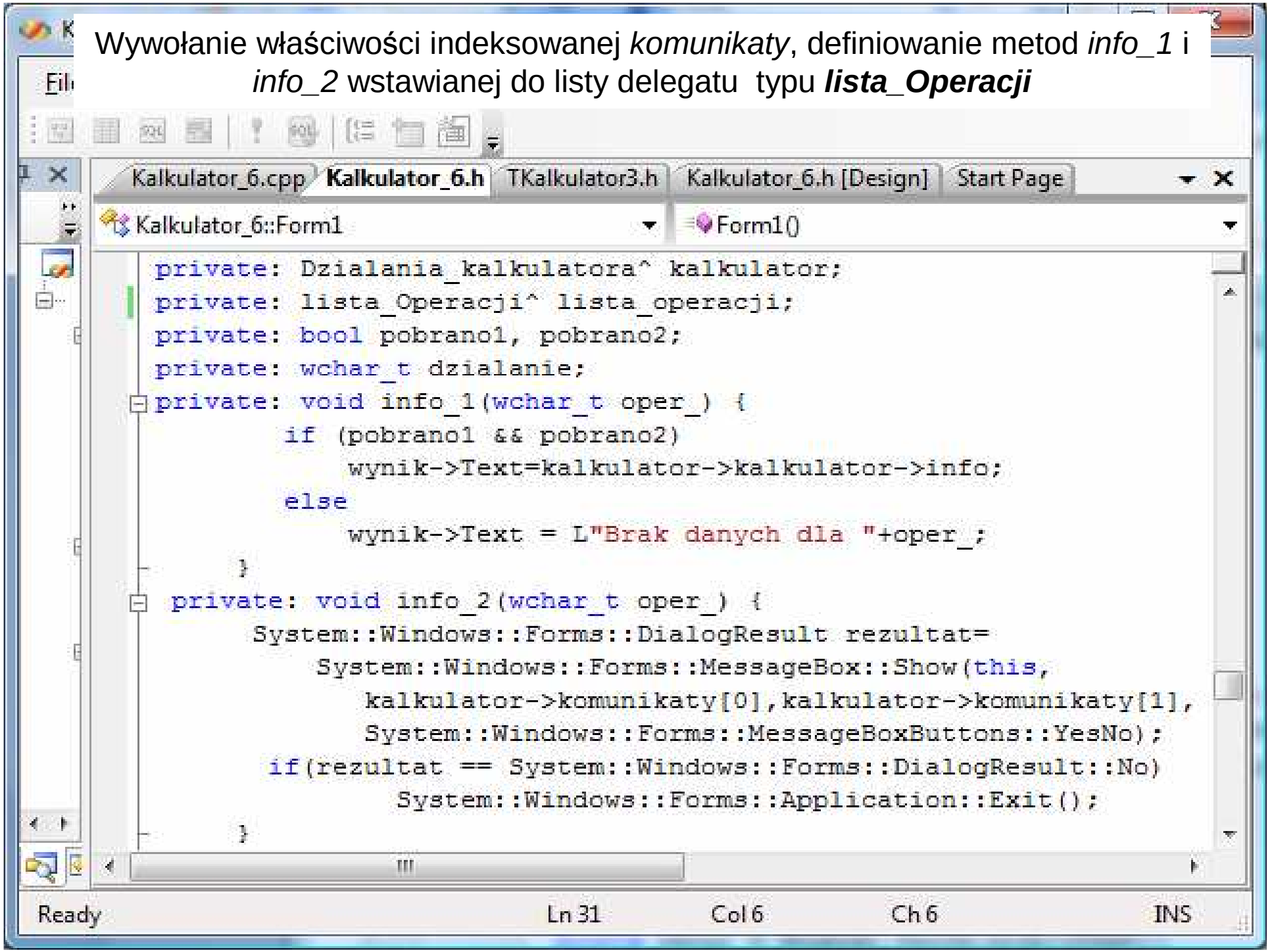
Zastosowanie klasy z uniwersalnym wykorzystaniem kalkulatora z GUI – **wersja 1 konstruktora** (1-y sposób tworzenia listy funkcji). Tworzenie listy pozwala tworzyć różne scenariusze działania aplikacji podczas jej działania)

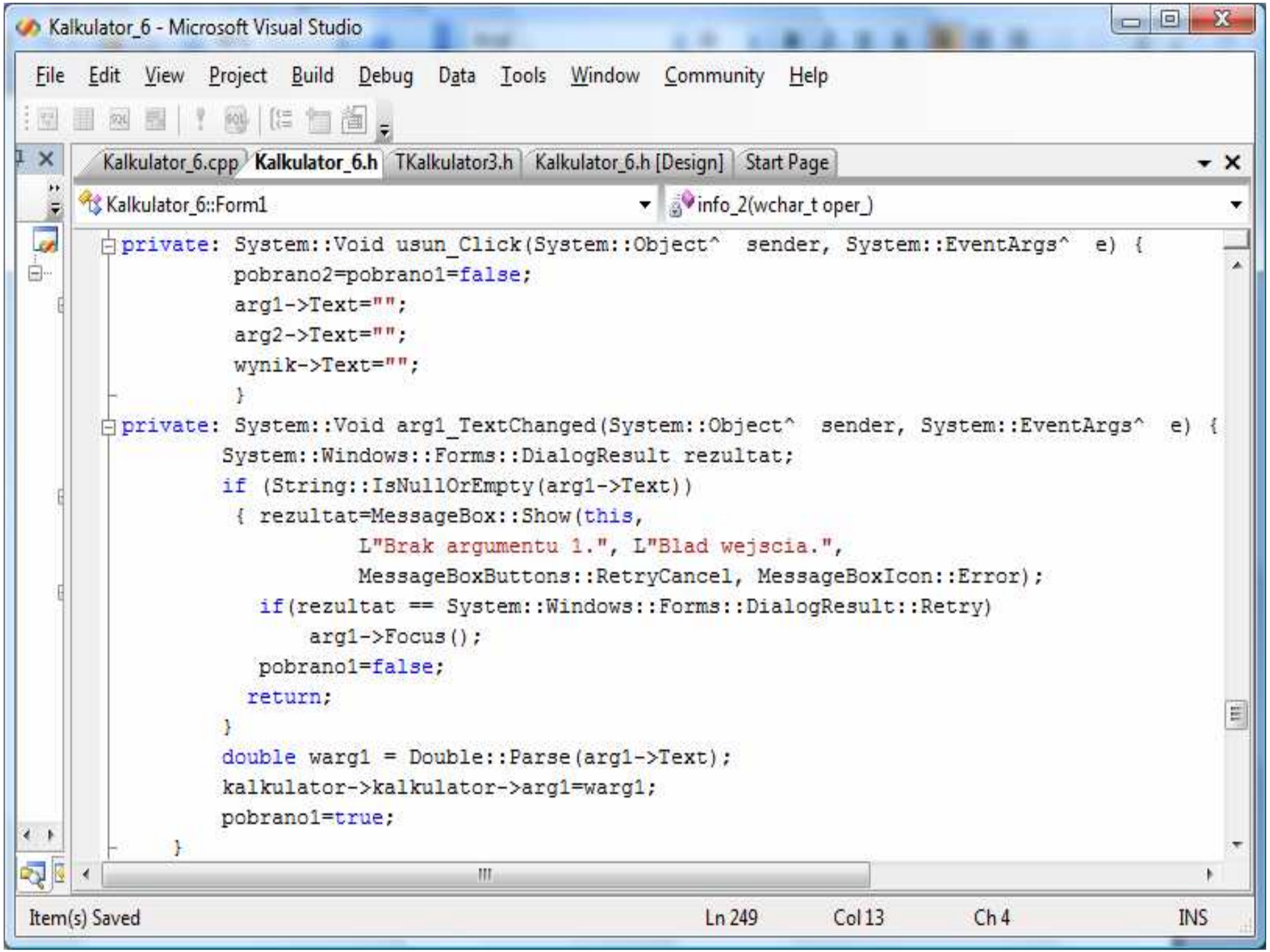


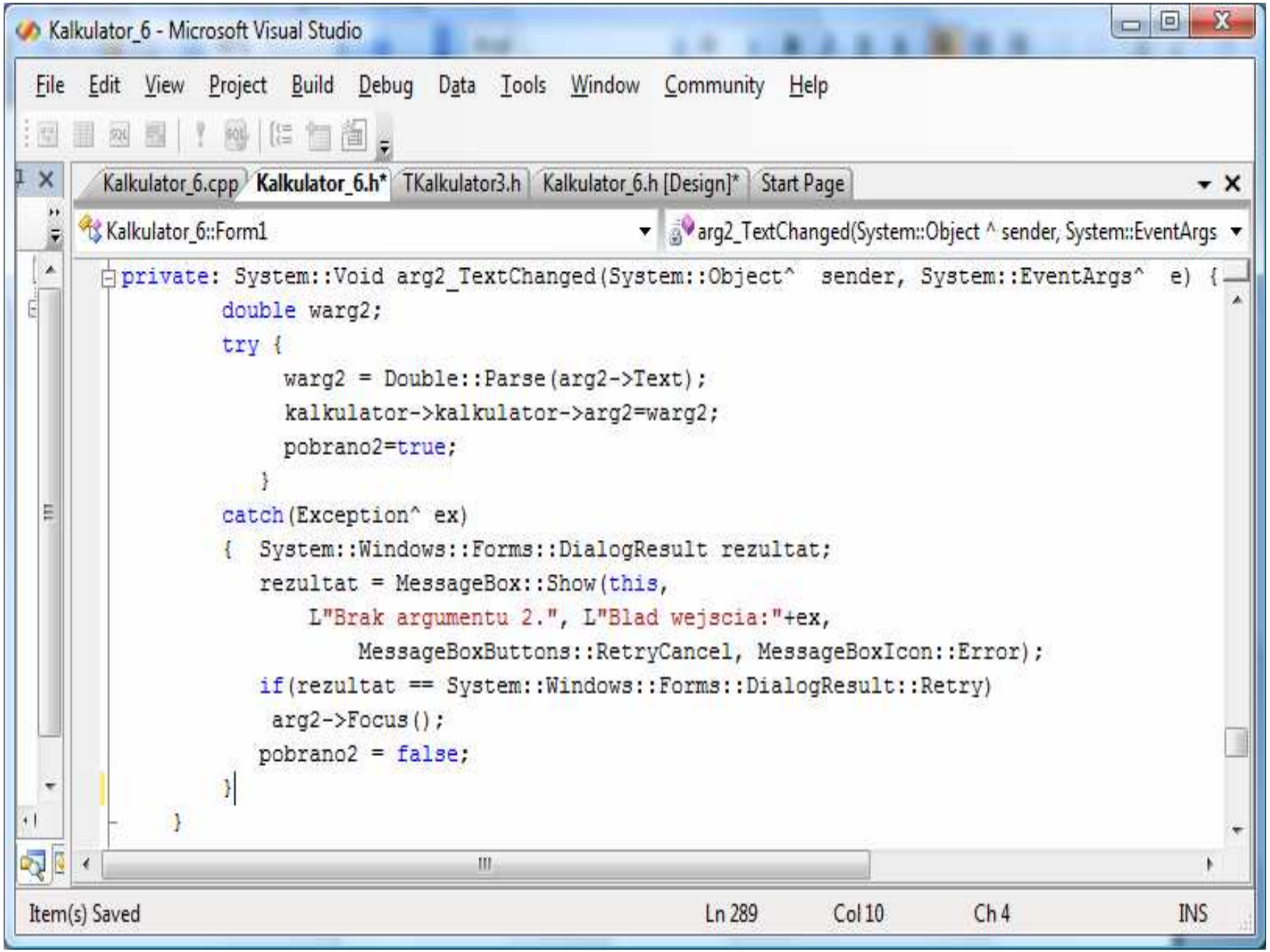
Zastosowanie klasy z uniwersalnym wykorzystaniem kalkulatora z GUI – **wersja 2 konstruktora** (2-i sposób tworzenia listy funkcji)

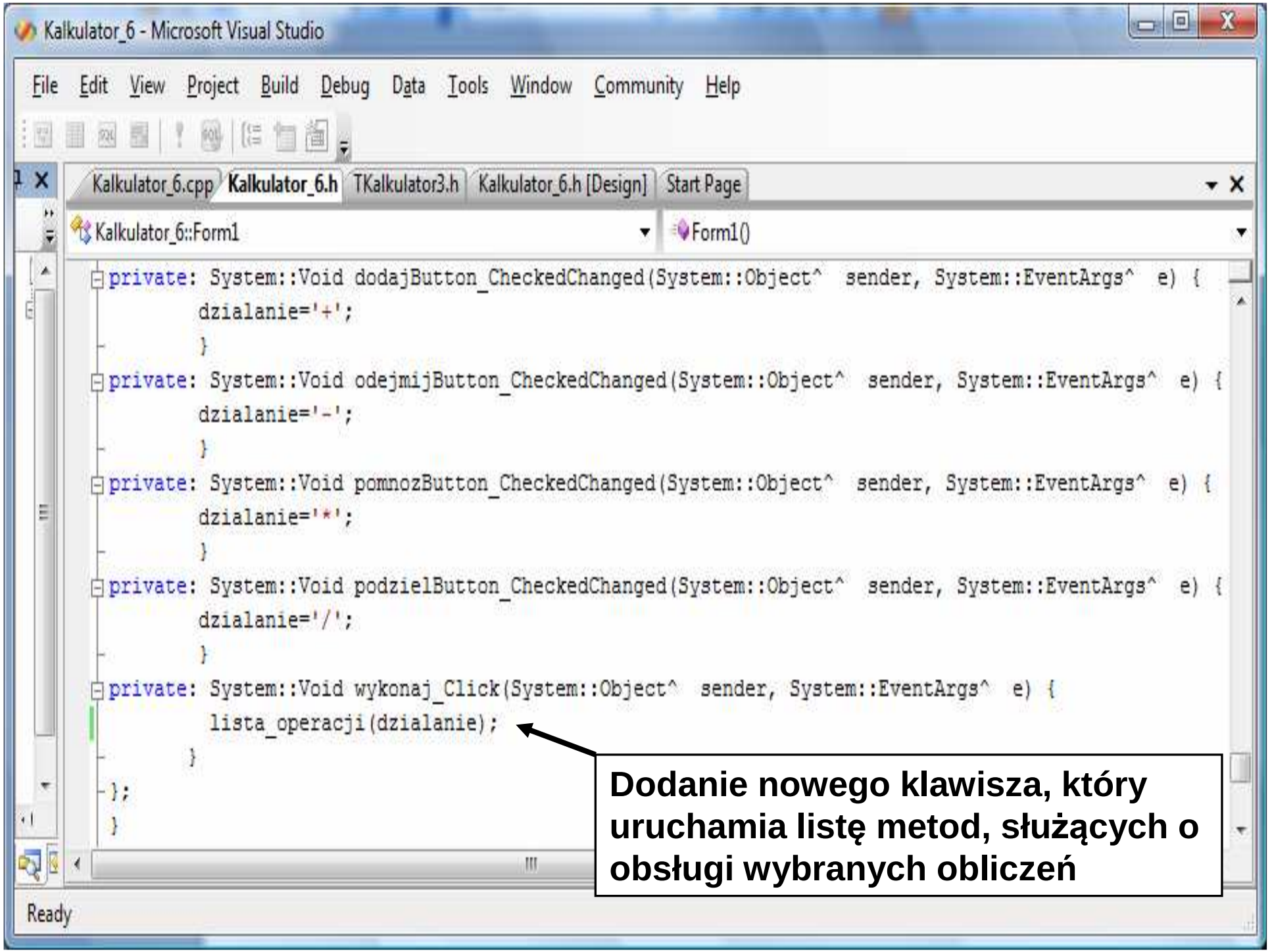


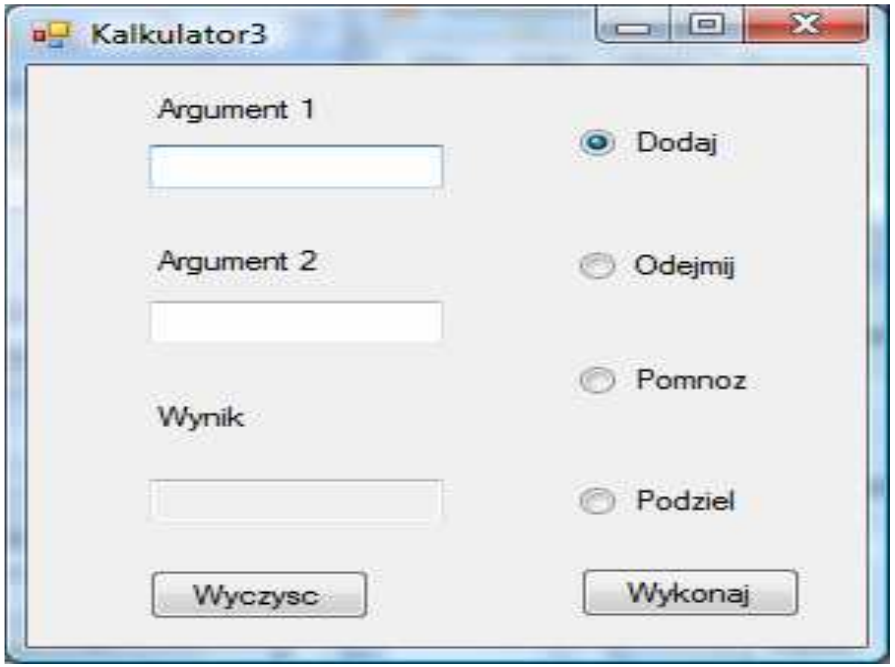
Wywołanie właściwości indeksowanej *komunikaty*, definiowanie metod *info_1* i *info_2* wstawianej do listy delegatu typu ***lista_Operacji***



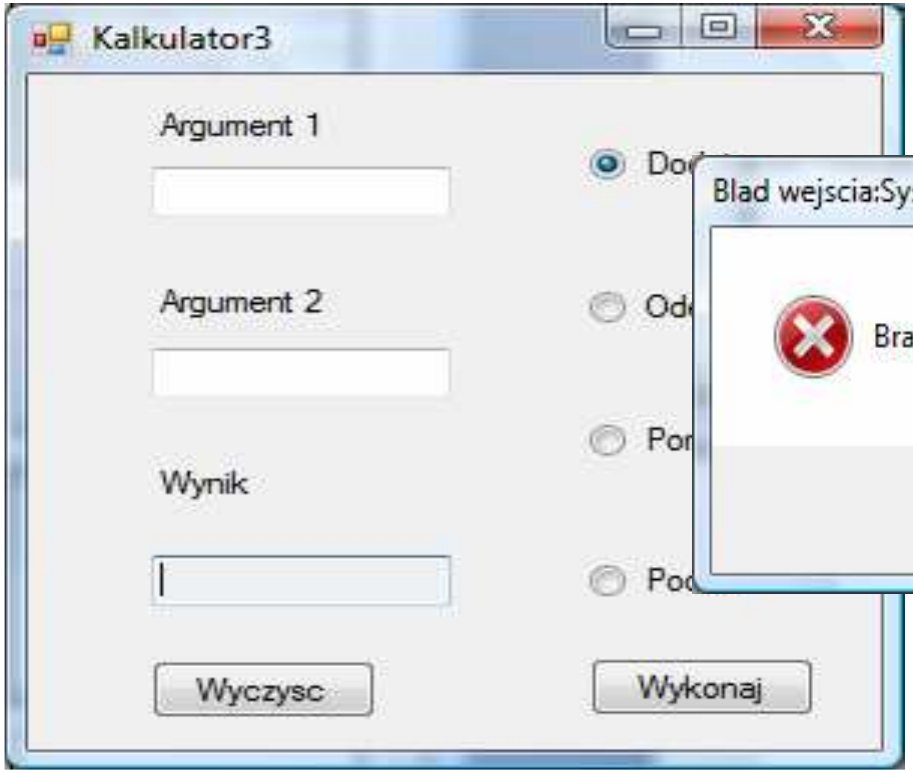

Kalkulator_6 - Microsoft Visual Studio
File Edit View Project Build Debug Data Tools Window Community Help
Kalkulator_6.cpp
Kalkulator_6.h
TKalkulator3.h
Kalkulator_6.h [Design]
Start Page
Kalkulator_6::Form1
info_2(wchar_t oper_)
private: System::Void usun_Click(System::Object^ sender, System::EventArgs^ e) {
pobrano2=pobrano1=false;
arg1->Text="";
arg2->Text="";
wynik->Text="";
}
private: System::Void arg1_TextChanged(System::Object^ sender, System::EventArgs^ e) {
System::Windows::Forms::DialogResult rezultat;
if (String::IsNullOrEmpty(arg1->Text))
{ rezultat=MessageBox::Show(this,
L"Brak argumentu 1.", L"Blad wejscia.",
MessageBoxButtons::RetryCancel, MessageBoxIcon::Error);
if(rezultat == System::Windows::Forms::DialogResult::Retry)
arg1->Focus();
pobrano1=false;
return;
}
double warg1 = Double::Parse(arg1->Text);
kalkulator->kalkulator->arg1=warg1;
pobrano1=true;
}
Item(s) Saved
Ln 249
Col 13
Ch 4
INS

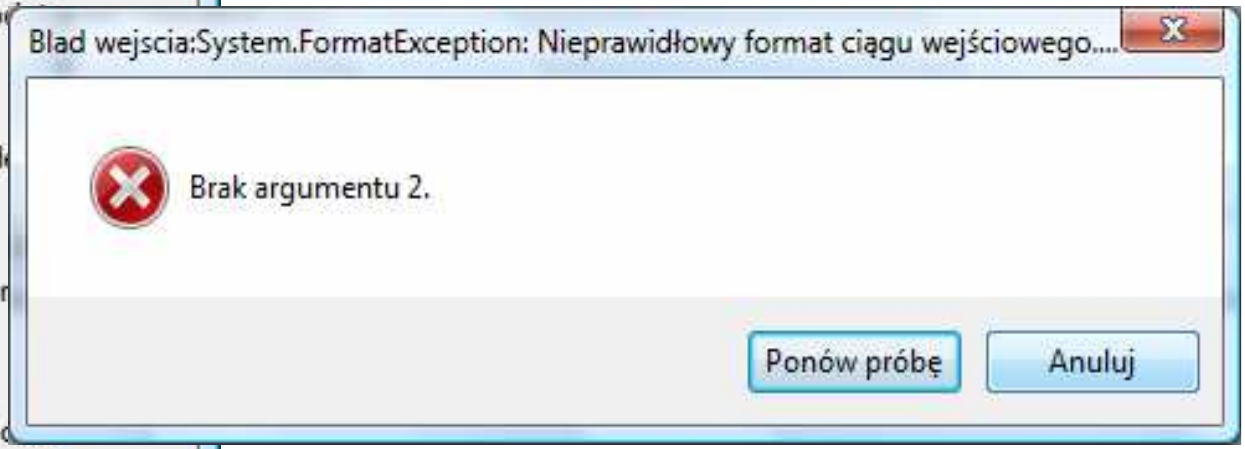

Kalkulator_6 - Microsoft Visual Studio
File Edit View Project Build Debug Data Tools Window Community Help
Kalkulator_6.cpp
Kalkulator_6.h*
TKalkulator3.h
Kalkulator_6.h [Design]*
Start Page
Kalkulator_6::Form1
arg2_TextChanged(System::Object ^ sender, System::EventArgs
private: System::Void arg2_TextChanged(System::Object^ sender, System::EventArgs^ e) {
double warg2;
try {
warg2 = Double::Parse(arg2->Text);
kalkulator->kalkulator->arg2=warg2;
pobrano2=true;
}
catch(Exception^ ex)
{ System::Windows::Forms::DialogResult rezultat;
rezultat = MessageBox::Show(this,
L"Brak argumentu 2.", L"Blad wejscia:"+ex,
MessageBoxButtons::RetryCancel, MessageBoxIcon::Error);
if(rezultat == System::Windows::Forms::DialogResult::Retry)
arg2->Focus();
pobrano2 = false;
}
}
Item(s) Saved
Ln 289
Col 10
Ch 4
INS


Kalkulator_6 - Microsoft Visual Studio
File Edit View Project Build Debug Data Tools Window Community Help
Kalkulator_6.cpp Kalkulator_6.h TKalkulator3.h Kalkulator_6.h [Design] Start Page
Kalkulator_6::Form1
Form1()
private: System::Void dodajButton_CheckedChanged(System::Object^ sender, System::EventArgs^ e) {
dzialanie='+';
}
private: System::Void odejmijButton_CheckedChanged(System::Object^ sender, System::EventArgs^ e) {
dzialanie='-';
}
private: System::Void pomnozButton_CheckedChanged(System::Object^ sender, System::EventArgs^ e) {
dzialanie='*';
}
private: System::Void podzielButton_CheckedChanged(System::Object^ sender, System::EventArgs^ e) {
dzialanie='/';
}
private: System::Void wykonaj_Click(System::Object^ sender, System::EventArgs^ e) {
lista_operacji(dzialanie);
}
};
}
Dodanie nowego klawisza, który uruchamia listę metod, służących o obsługi wybranych obliczeń
Ready

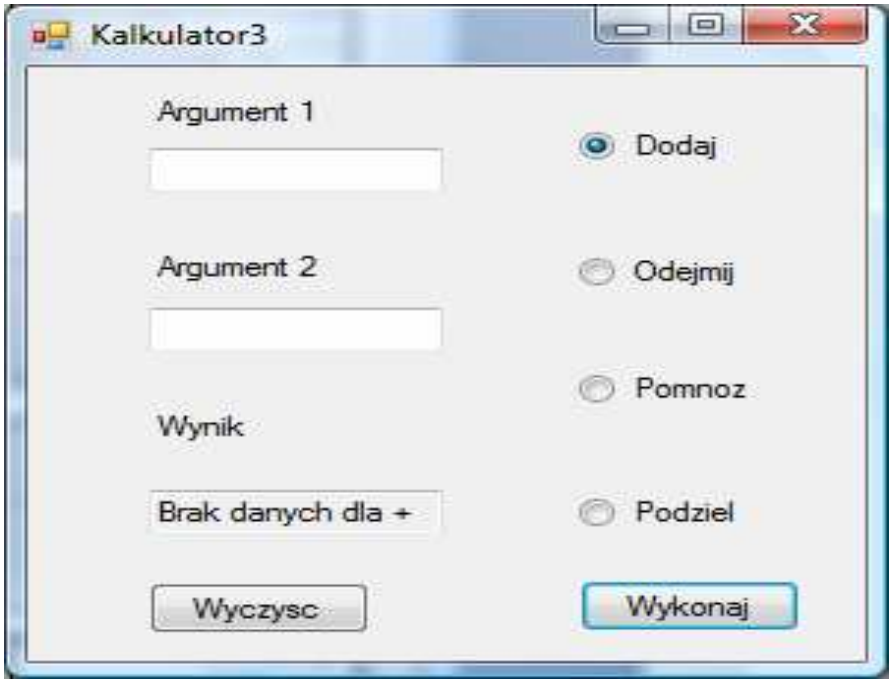

Kalkulator3
Argument 1
Dodaj
Argument 2
Odejmij
Pomnoz
Wynik
Podziel
Wyczysc
Wykonaj

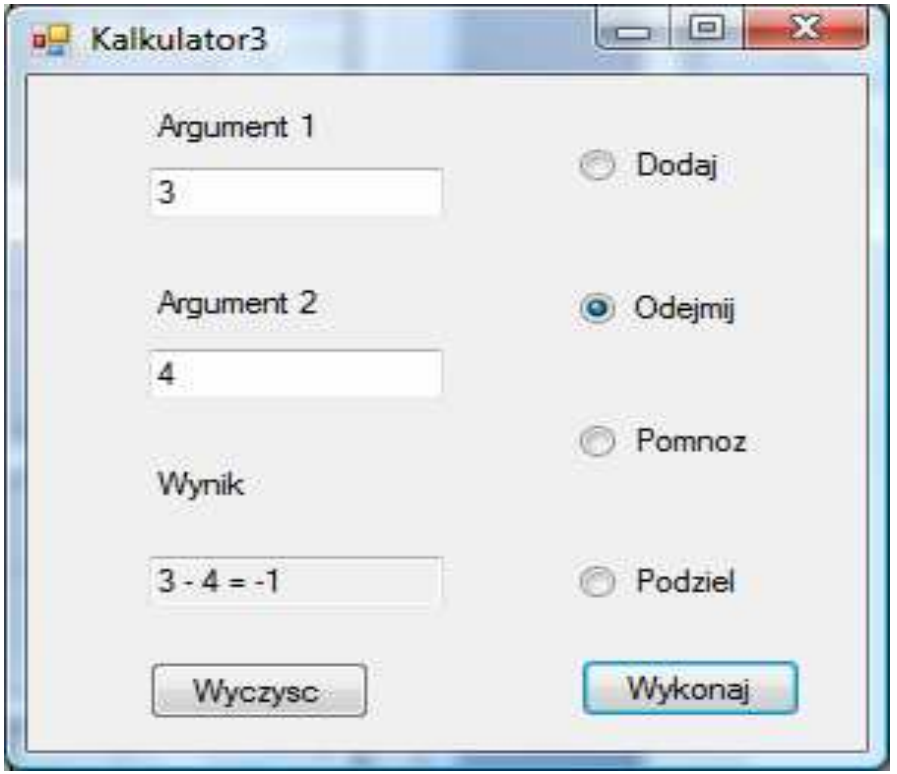

Blad wejscia.
Brak argumentu 1.
Ponów próbę
Anuluj

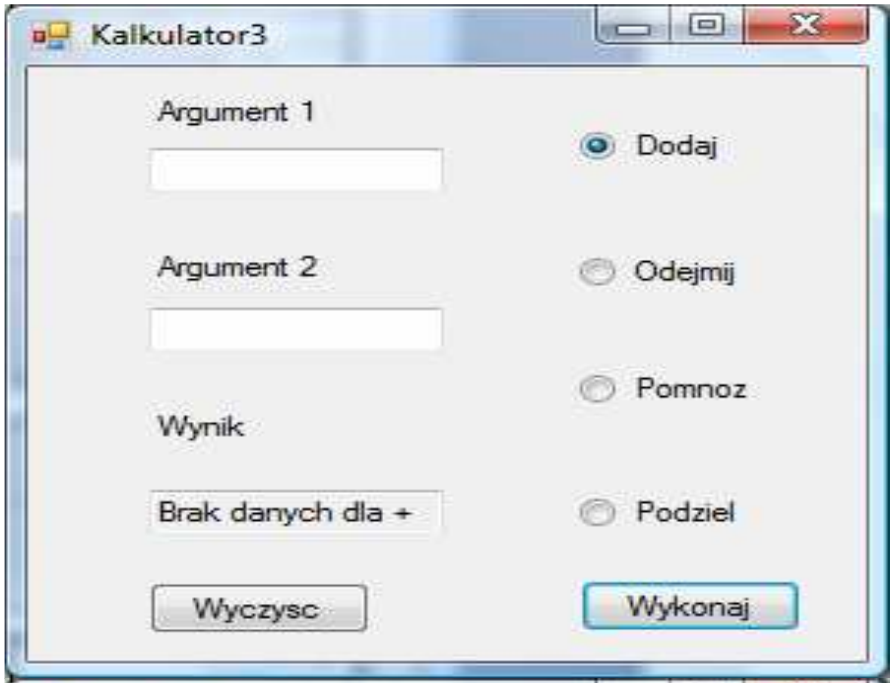

Kalkulator3
Argument 1
Argument 2
Wynik
Wyczysc
Wykonaj

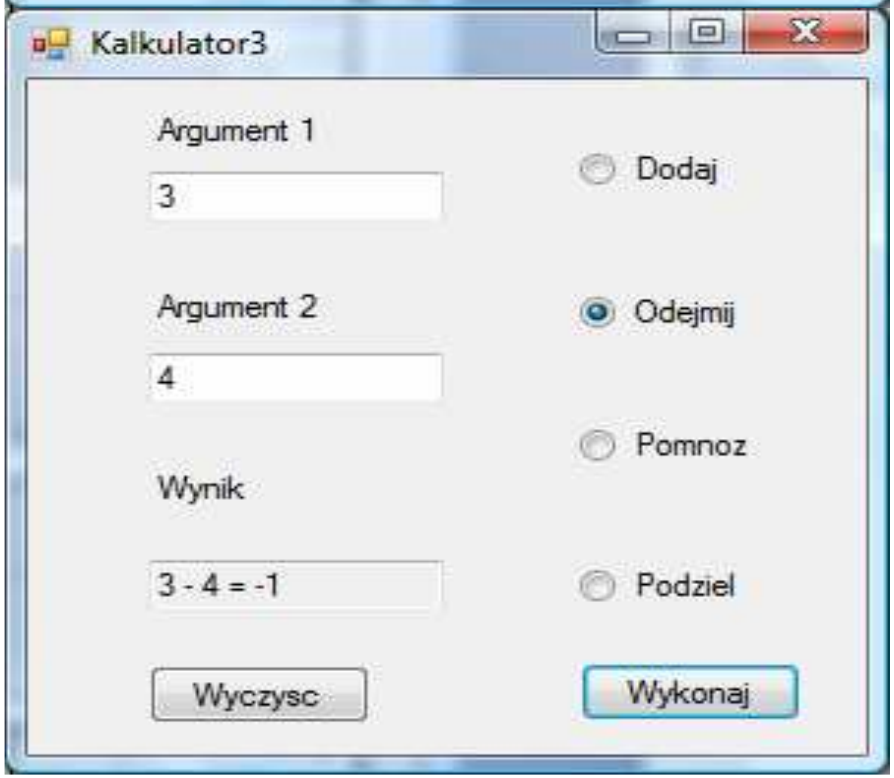

Blad wejscia:System.FormatException: Nieprawidłowy format ciągu wejściowego....
Brak argumentu 2.
Ponów próbę
Anuluj

Pierwszy scenariusz działania aplikacji (**konstruktor 1** z pierwszą listą)- zamykanie aplikacji za pomocą prawego górnego klawisza ze znakiem „x”

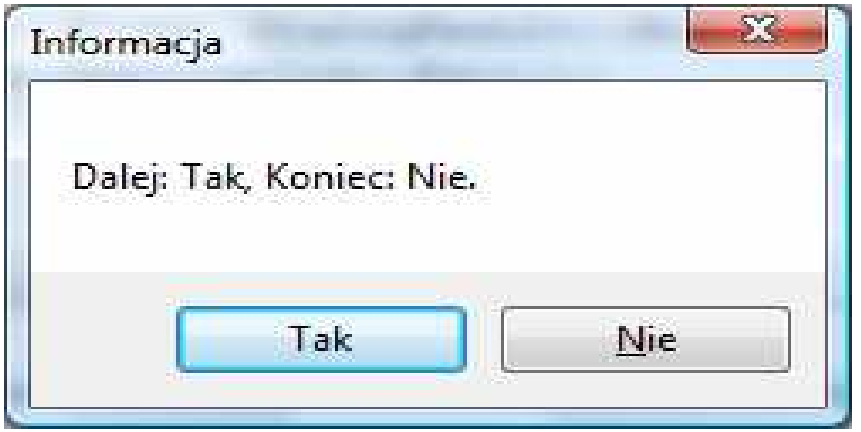


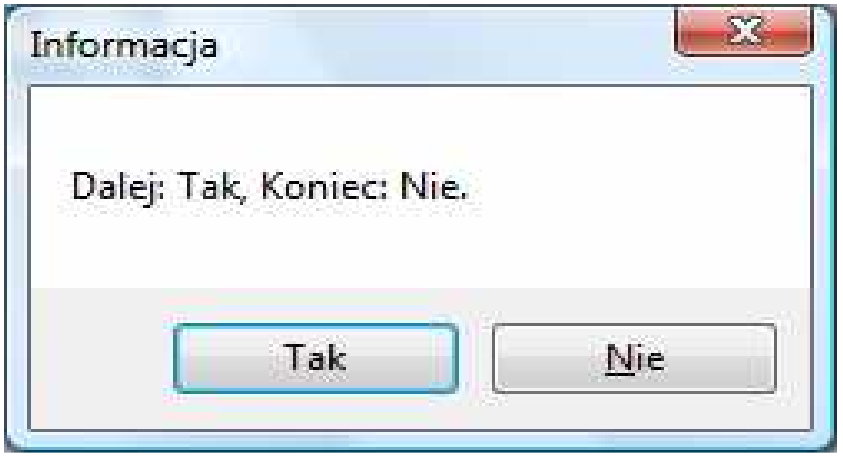




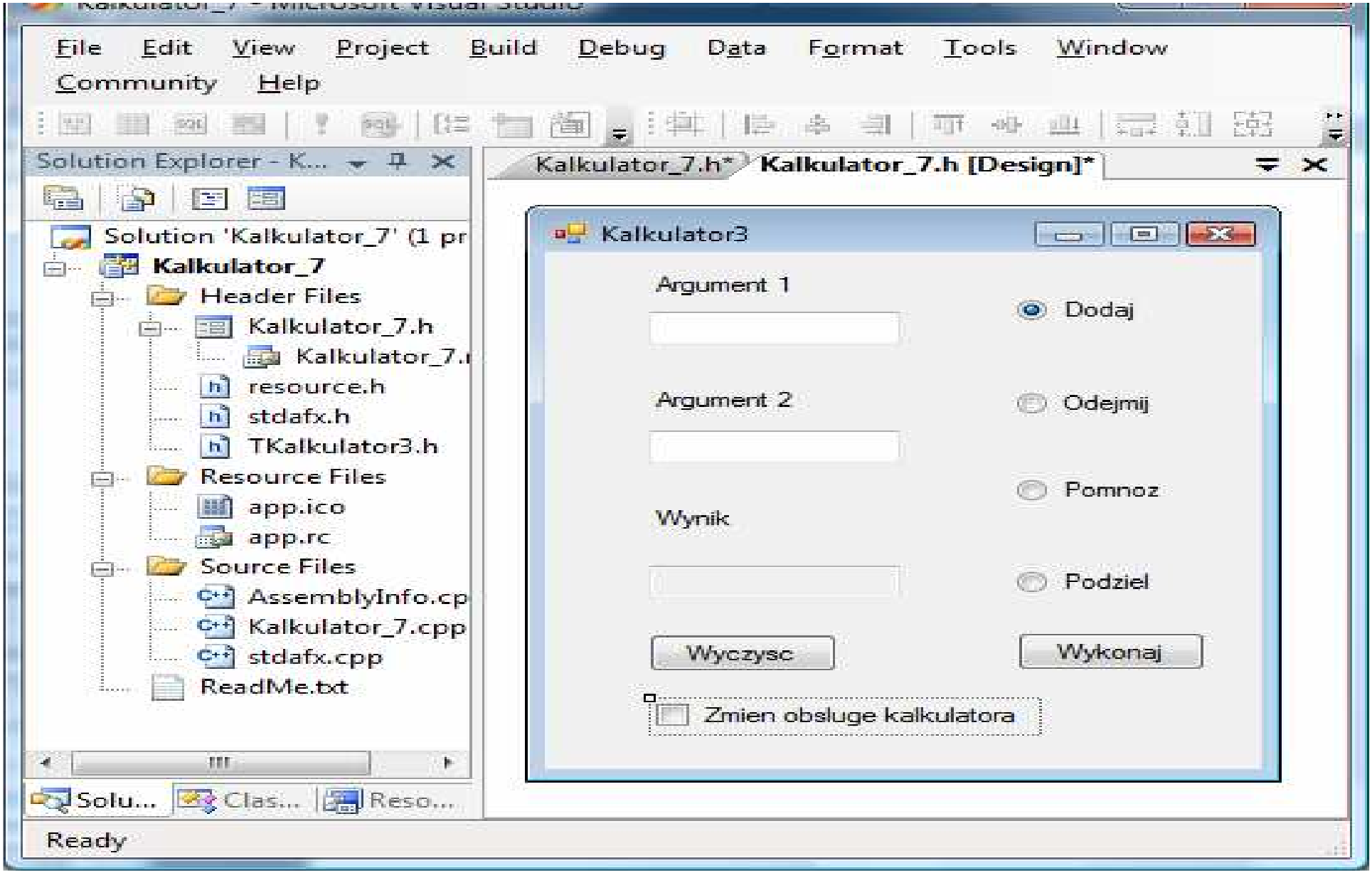


Drugi scenariusz działania aplikacji (**konstruktor 2** z drugą listą)- zamykanie aplikacji za pomocą prawego górnego klawisza ze znakiem „x” lub przycisku „Nie” z okienka dialogowego. Naciśnięcie klawisza „Tak” powoduje powrót do okna działań kalkulatora. Okienko dialogowe pojawia się po naciśnięciu klawisza „Wykonaj”

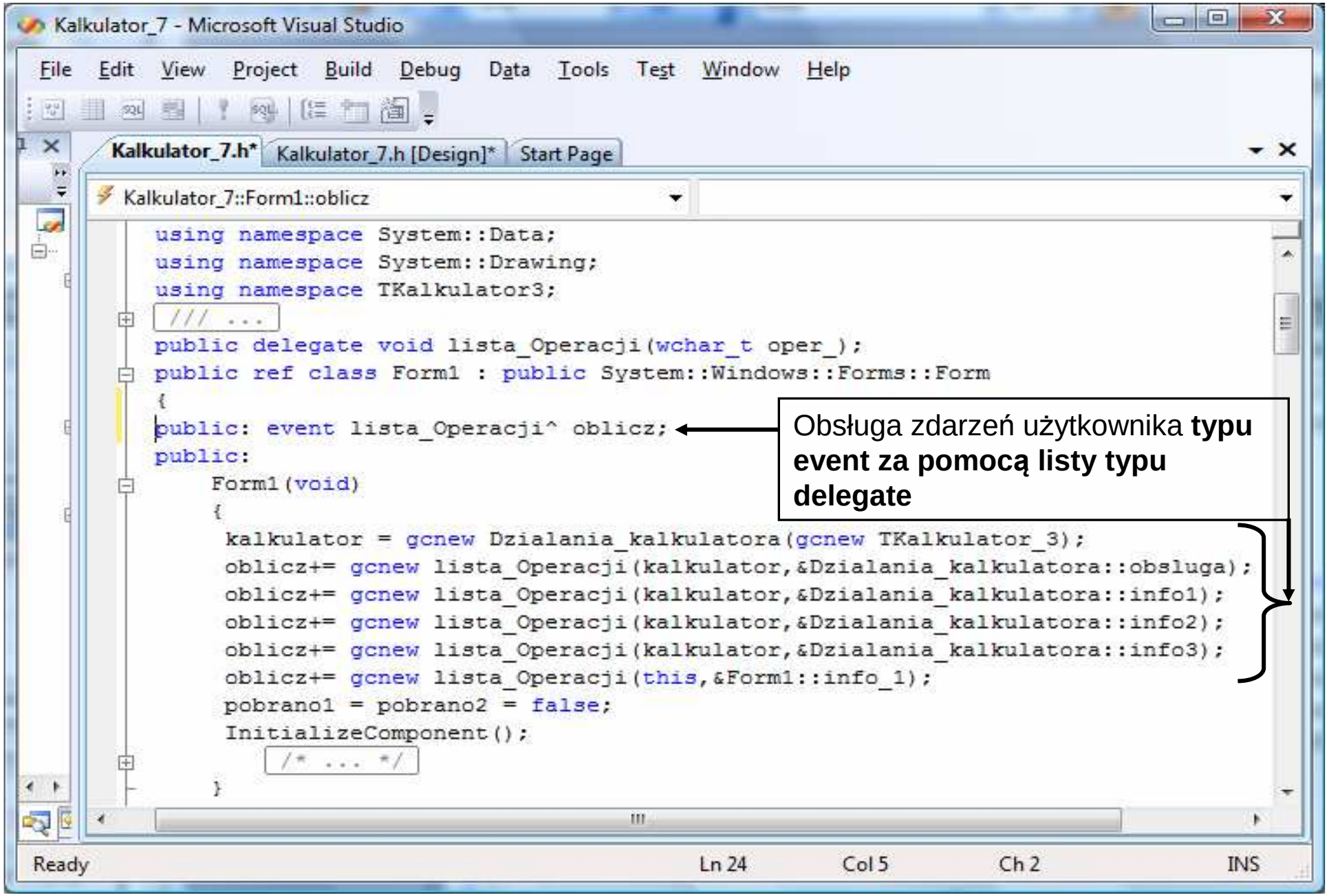




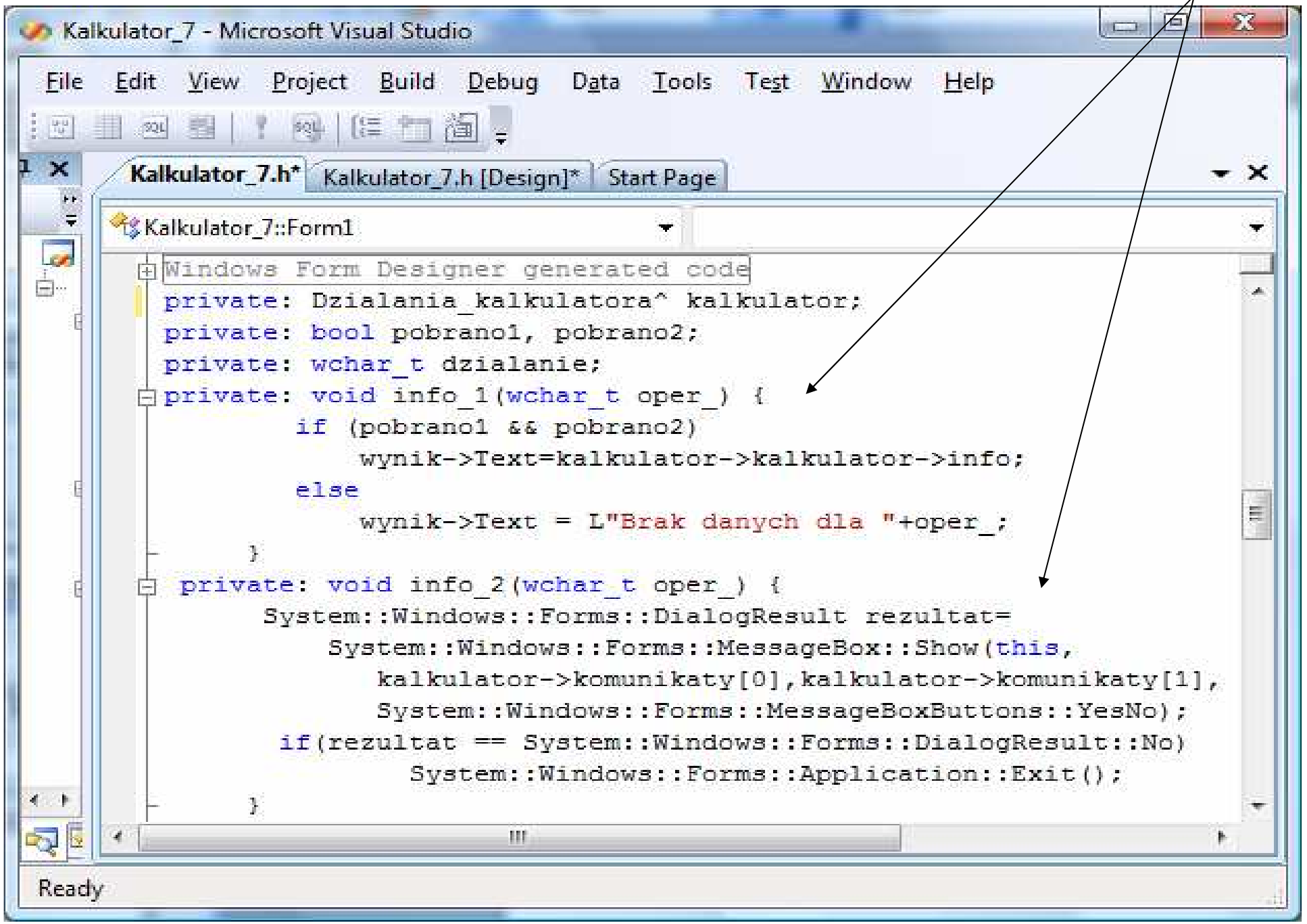
4. Zastosowanie delegatów do obsługi zdarzeń użytkownika

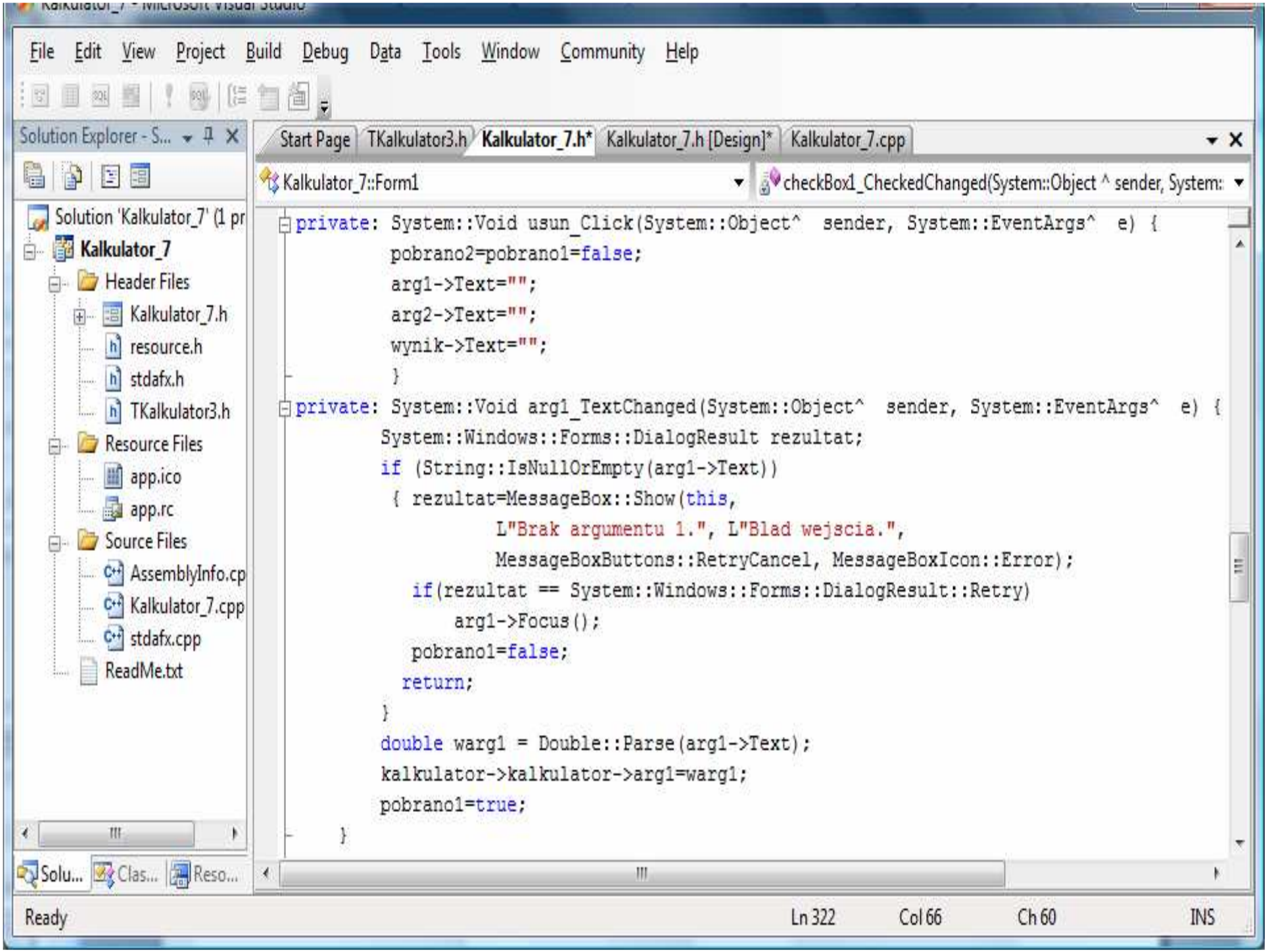


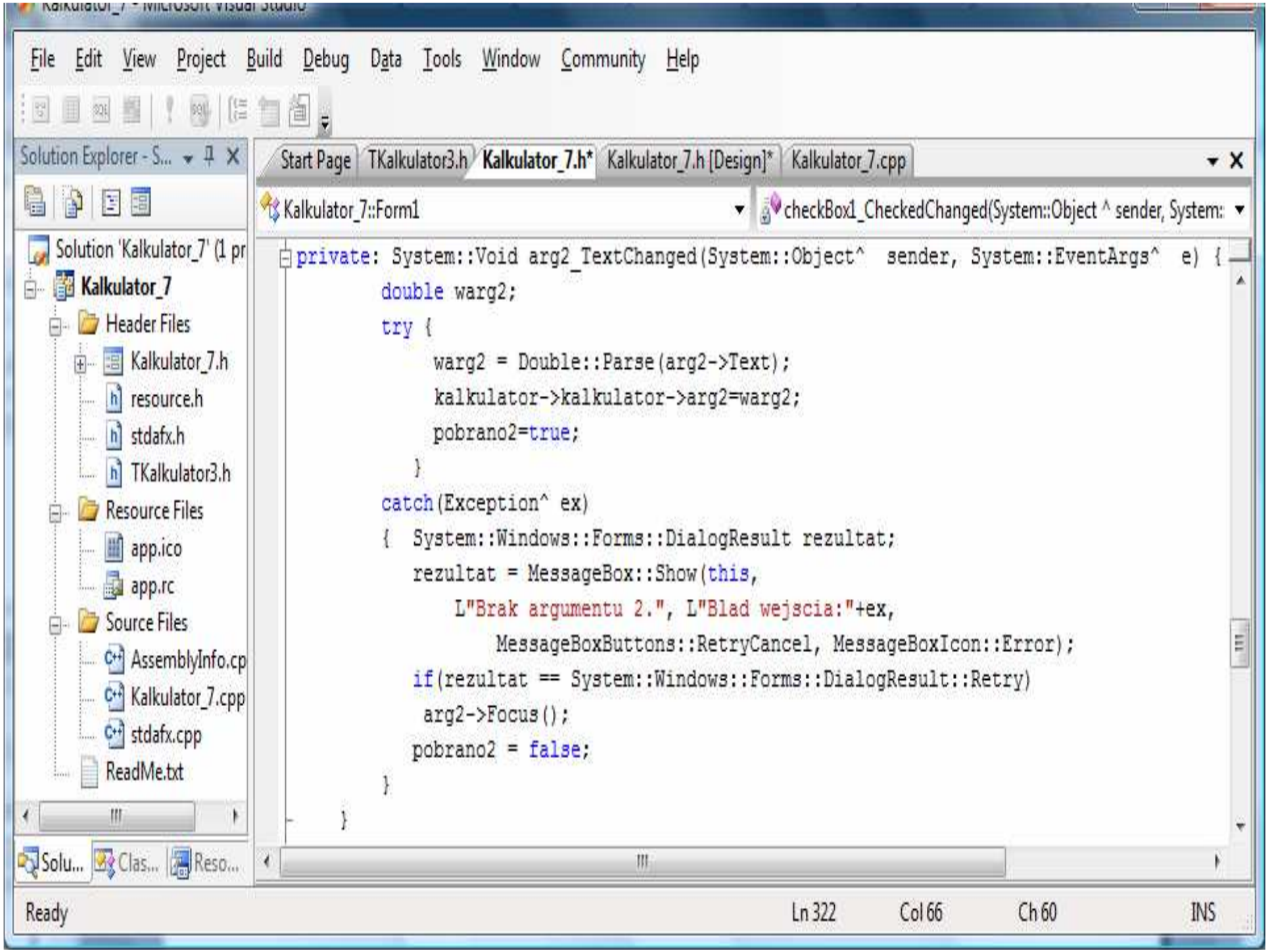
Okno Kalkulator3 domyślnie zamykane jest przyciśnięciem górnego prawego przycisku oznaczonego znakiem '+'

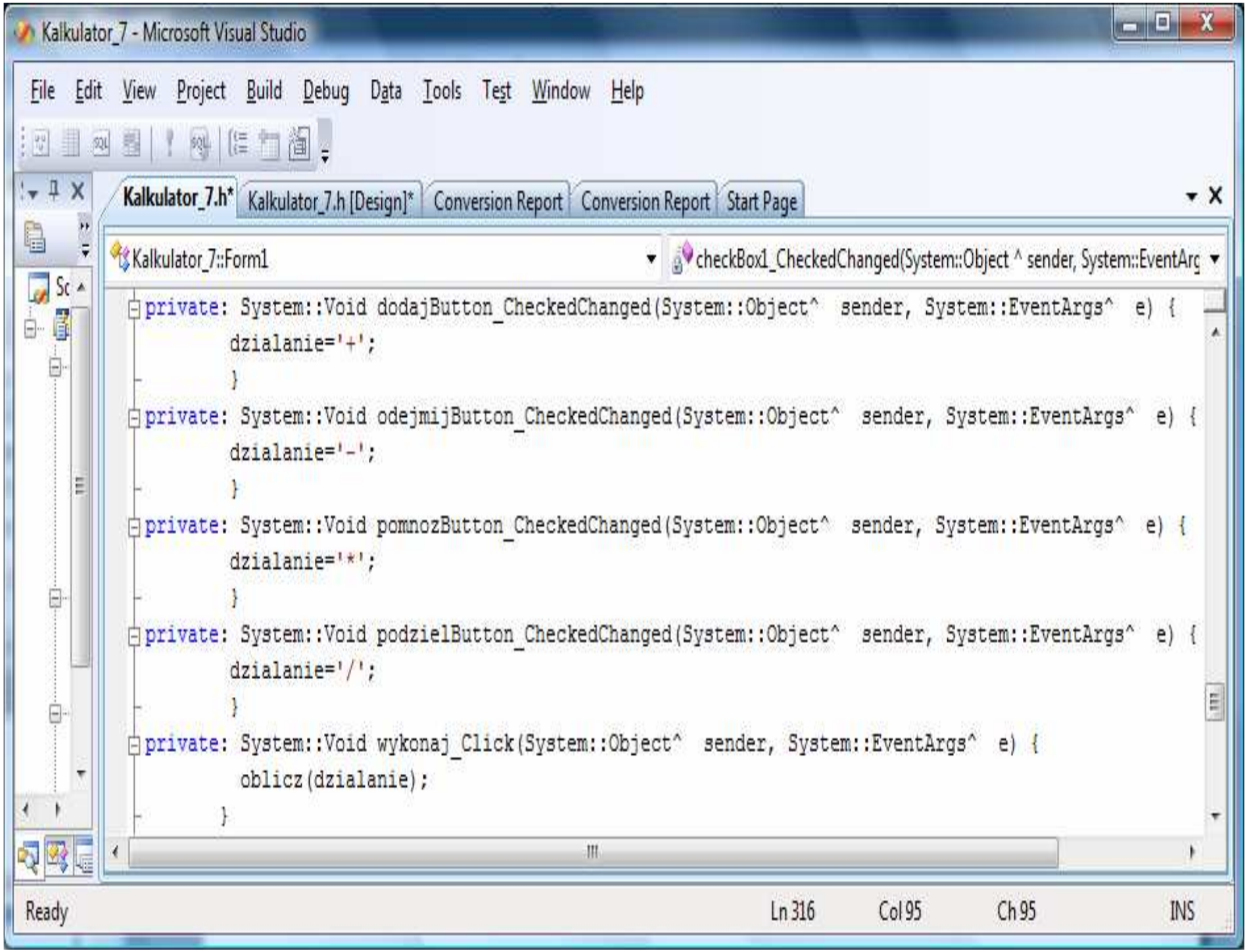


Funkcje do obsługi zdarzeń jako elementy listy typu delegate

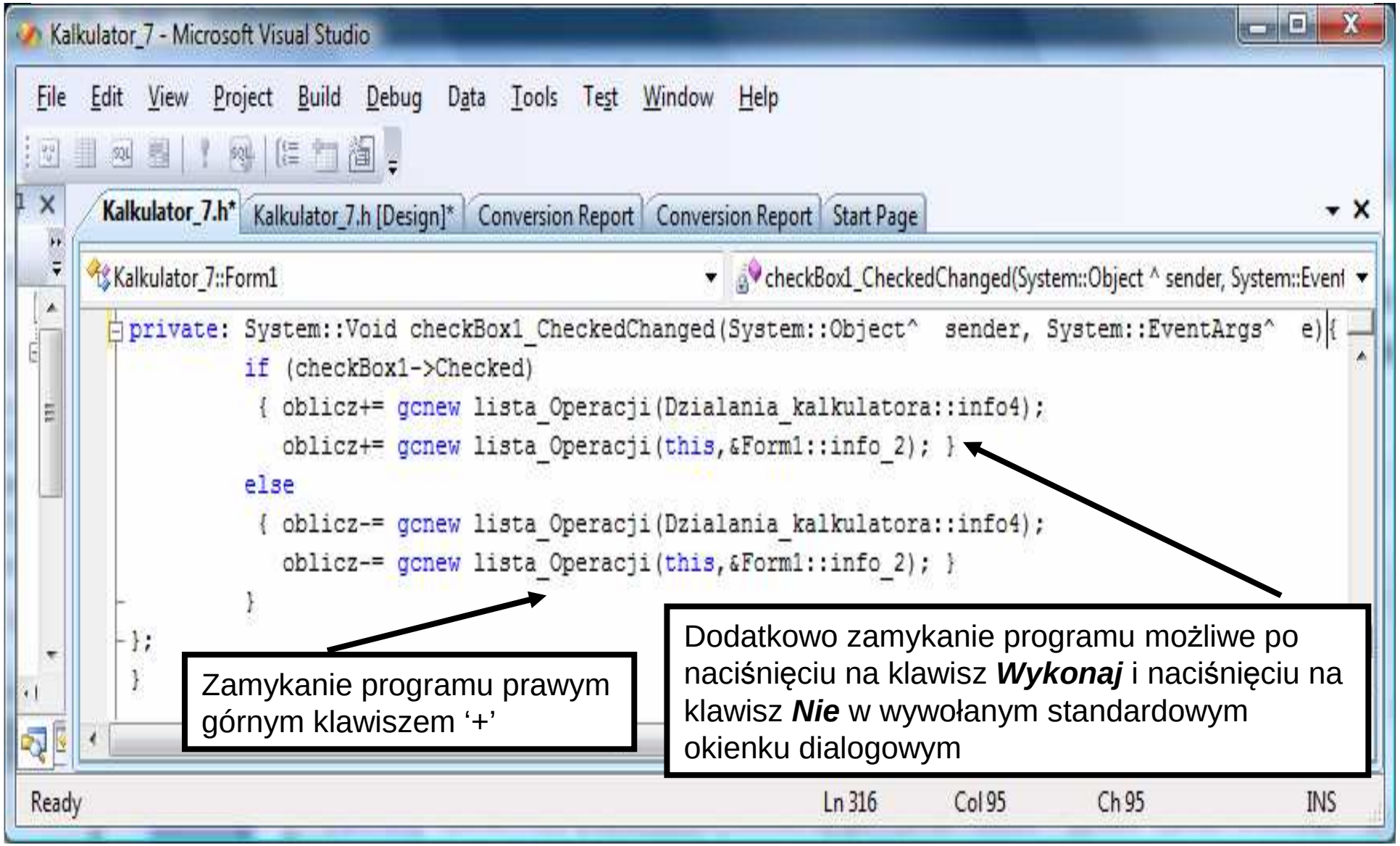



File Edit View Project Build Debug Data Tools Window Community Help
Solution Explorer - S...
Solution 'Kalkulator_7' (1 pr
Kalkulator_7
Header Files
Kalkulator_7.h
resource.h
stdafx.h
TKalkulator3.h
Resource Files
app.ico
app.rc
Source Files
AssemblyInfo.cp
Kalkulator_7.cpp
stdafx.cpp
ReadMe.txt
Start Page
TKalkulator3.h
Kalkulator_7.h*
Kalkulator_7.h [Design]*
Kalkulator_7.cpp
Kalkulator_7::Form1
checkBox1_CheckedChanged(System::Object ^ sender, System:
private: System::Void usun_Click(System::Object^ sender, System::EventArgs^ e) {
pobrano2=pobrano1=false;
arg1->Text="";
arg2->Text="";
wynik->Text="";
}
private: System::Void arg1_TextChanged(System::Object^ sender, System::EventArgs^ e) {
System::Windows::Forms::DialogResult rezultat;
if (String::IsNullOrEmpty(arg1->Text))
{ rezultat=MessageBox::Show(this,
L"Brak argumentu 1.", L"Blad wejscia.",
MessageBoxButtons::RetryCancel, MessageBoxIcon::Error);
if(rezultat == System::Windows::Forms::DialogResult::Retry)
arg1->Focus();
pobrano1=false;
return;
}
double warg1 = Double::Parse(arg1->Text);
kalkulator->kalkulator->arg1=warg1;
pobrano1=true;
}
Solu...
Clas...
Reso...
Ready
Ln 322
Col 66
Ch 60
INS

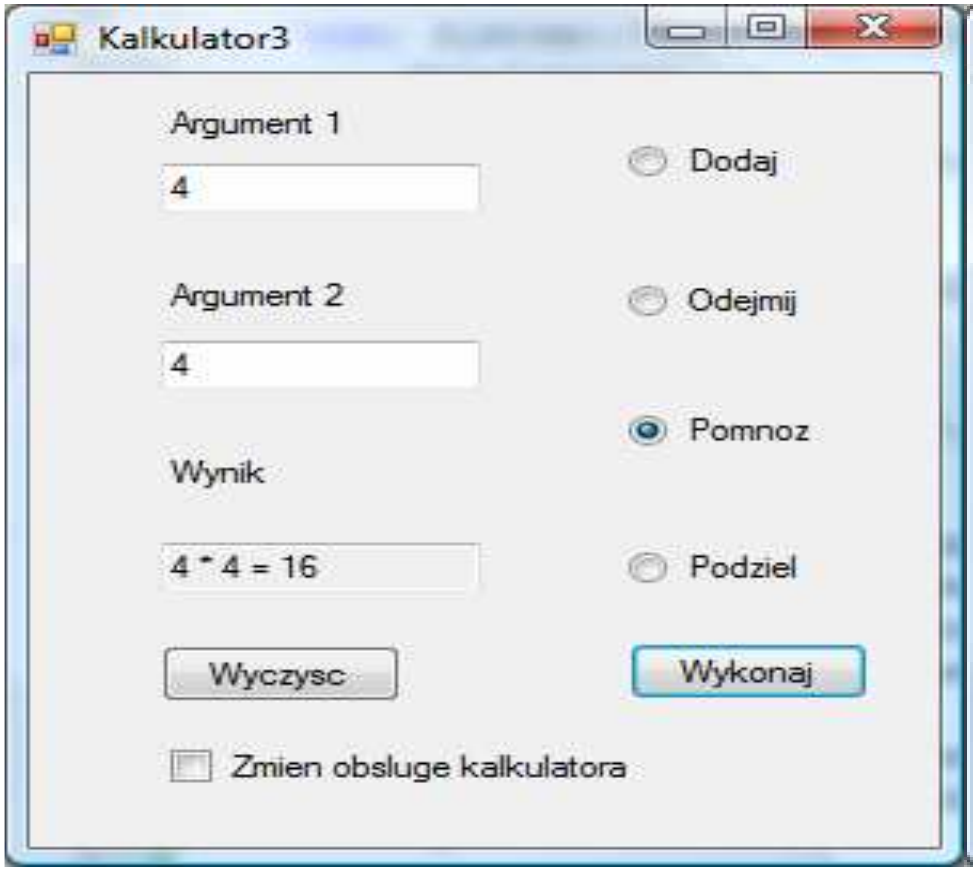

File Edit View Project Build Debug Data Tools Window Community Help
Solution Explorer - S...
Start Page
TKalkulator3.h
Kalkulator_7.h*
Kalkulator_7.h [Design]*
Kalkulator_7.cpp
Kalkulator_7::Form1
checkBox1_CheckedChanged(System::Object ^ sender, System:
Solution 'Kalkulator_7' (1 pr
Kalkulator_7
Header Files
Kalkulator_7.h
resource.h
stdafx.h
TKalkulator3.h
Resource Files
app.ico
app.rc
Source Files
AssemblyInfo.cp
Kalkulator_7.cpp
stdafx.cpp
ReadMe.txt
private: System::Void arg2_TextChanged(System::Object^ sender, System::EventArgs^ e) {
double warg2;
try {
warg2 = Double::Parse(arg2->Text);
kalkulator->kalkulator->arg2=warg2;
pobrano2=true;
}
catch(Exception^ ex)
{ System::Windows::Forms::DialogResult rezultat;
rezultat = MessageBox::Show(this,
L"Brak argumentu 2.", L"Blad wejscia:"+ex,
MessageBoxButtons::RetryCancel, MessageBoxIcon::Error);
if(rezultat == System::Windows::Forms::DialogResult::Retry)
arg2->Focus();
pobrano2 = false;
}
}
Solu...
Clas...
Reso...
Ready
Ln 322
Col 66
Ch 60
INS


Kalkulator_7 - Microsoft Visual Studio
File Edit View Project Build Debug Data Tools Test Window Help
Kalkulator_7.h*
Kalkulator_7.h [Design]*
Conversion Report
Conversion Report
Start Page
Kalkulator_7::Form1
checkBox1_CheckedChanged(System::Object ^ sender, System::EventArg
private: System::Void dodajButton_CheckedChanged(System::Object^ sender, System::EventArgs^ e) {
dzialanie='+';
}
private: System::Void odejmijButton_CheckedChanged(System::Object^ sender, System::EventArgs^ e) {
dzialanie='-';
}
private: System::Void pomnozButton_CheckedChanged(System::Object^ sender, System::EventArgs^ e) {
dzialanie='*';
}
private: System::Void podzielButton_CheckedChanged(System::Object^ sender, System::EventArgs^ e) {
dzialanie='/';
}
private: System::Void wykonaj_Click(System::Object^ sender, System::EventArgs^ e) {
oblicz(dzialanie);
}
Ready
Ln 316
Col 95
Ch 95
INS

Zaznaczanie przycisku typu **CheckBox** (obsługa metodą ***checkBox1_CheckedChanged***) pozwala zmieniać obsługę zdarzenia klikania na przycisk ***Wykonaj*** (zmiana scenariusza zdarzenia ***oblicz*** wywołanego jako wtórne zdarzenie w metodzie ***wykonaj_Click***)



Pierwszy scenariusz działania kalkulatora



Drugi scenariusz działania kalkulatora