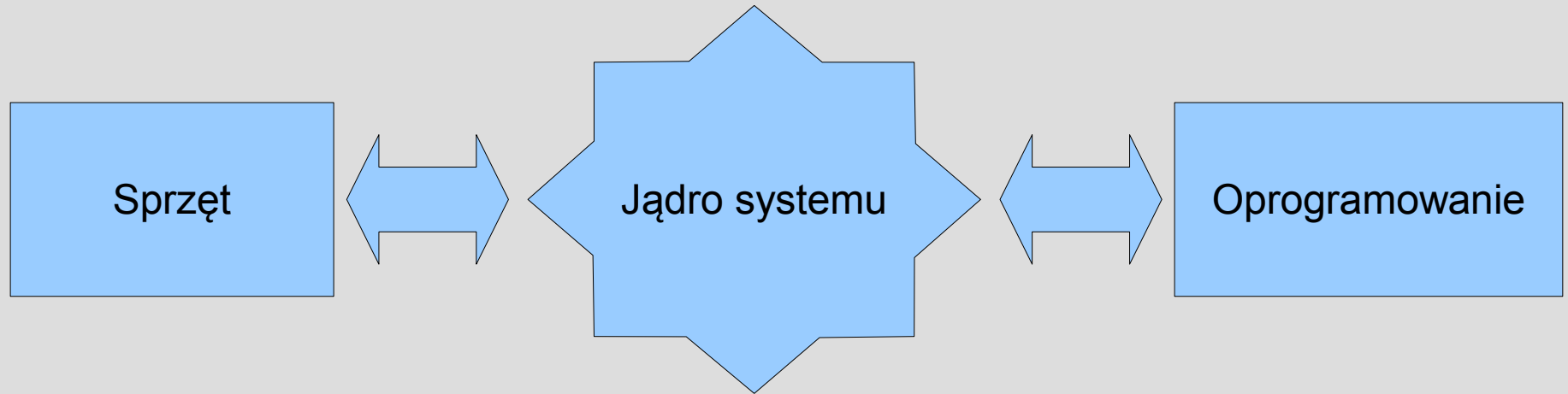


Języki i metodyka programowania

Obiektowość systemów operacyjnych – Windows.

Języki i metodyka programowania

Obiektowość systemów operacyjnych – Windows



Języki i metodyka programowania

Obiektowość systemów operacyjnych – Windows

Akcje powiązane z komunikatami:

MouseMove

MouseOver

MouseLeave

MouseDown

MouseUp

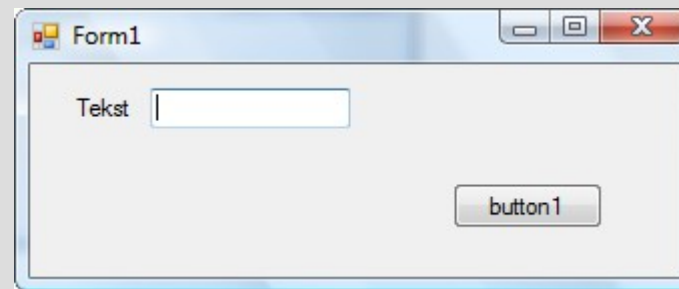
MouseDoubleClick

MouseClick

KeyDown

KeyPress

KeyUp



Click

Enter

Leave

TextChanged

...

Języki i metodyka programowania

**Wstęp do języka C#
obiekty i klasy.**

Języki i metodyka programowania

Wstęp do języka C#

Literatura:

- John Sharp, Jon Jagger „Microsoft Visual C#.NET”, Wydawnictwo RM
- The C# Programming language, 4th edition
<http://it-ebooks.info/book/916/>

Języki i metodyka programowania

Wstęp do języka C#

Microsoft .NET Framework – to platforma firmy Microsoft realizująca wizję umożliwiającą wykonywanie aplikacji dystrybuowanych w internecie niezależnie od systemu operacyjnego, na którym są one uruchamiane.

CLR – (Common Language Runtime) to środowisko uruchomieniowe wspólnego języka (klasy ASP.NET, ADO.NET, ...)

BCL – (Base Class Library) biblioteka dostępna dla wszystkich języków programowania używających platformy .NET. Dostarcza podstawowe funkcjonalności wykorzystywane w aplikacjach.

Języki i metodyka programowania

Wstęp do języka C#

Podobieństwa C# a C:

- Wielkość liter jest istotna,
- Instrukcja kończy się średnikiem,
- Podobne typy, operatory i wyrażenia.

Różnice C# a C:

- C jest językiem strukturalnym a C# obiekowym,
- W C# nie są stosowane zbiory nagłówkowe `#include<plik.h>`
Używa się przestrzeni nazw (Namespaces) `using System,`
- Programy w C# definiowane są w klasach,
- W C wartości parametrów odczytywane są bezpośrednio,
W C# parametry odczytywane są poprzez metody.

Języki i metodyka programowania

Język C# obiekty i klasy

Język C# jest obiektowy, dlatego podstawowym elementem tego języka jest obiekt.

Do modelowania obiektów wykorzystuje się klasy.

Każdy program w C# jest powiązany z określonym obiektem oraz implementowany wewnątrz klasy opisującej dany obiekt.

Języki i metodyka programowania

Język C# obiekty i klasy

Klasa – to (zgodnie z UML) element opisujący zbiór obiektów charakteryzujących się wspólnymi atrybutami i operacjami.

W C#:

Klasa – to element przedstawiający typ, a raczej składowe tego typu. Klasa zawiera zmienne (atrybuty) oraz metody (funkcje).

Nazwa
Atrybuty
Operacje

```
[<atrybuty>] [<modyfikator>] class <nazwa> <ciało>
```

Języki i metodyka programowania

Język C# obiekty i klasy

```
[<atrybuty>] [<modyfikator>] class <nazwa> [<klasa bazowa>] [, <interfejsy>] <ciało>
```

atrybuty pozwalają na dostarczenie do kompilatora .NET dodatkowych informacji (np. import klasy z biblioteki, serializacja)

modyfikatory

public	Typ dostępny dla innych typów.
internal	Typ widoczny tylko dla danego zbioru.
protected	Typ dostępny poprzez zawarte typy oraz dziedziczące.
protected internal	Typ stanowiący połączenie typu protected oraz internal
private	Dostęp do typu tylko dla właściciela.
new	Słowo kluczowe new może być używane dla klas zagnieżdżonych (klas zdefiniowanych wewnątrz innej klasy).
abstract	Klasa, która nie może być używana do definicji obiektu, a tylko poprzez dziedziczenie.
sealed	Klasa nie może być wykorzystywana w dziedziczeniu.

Języki i metodyka programowania

Język C# obiekty i klasy

```
[<atributy>] [<modyfikator>] class <nazwa> [<klasa bazowa>] [, <interfejsy>] <ciało>
```

klasa bazowa

definiowana jest po dwukropku i oznacza, iż dana klasa dziedziczy atrybuty i operacje klasy bazowej.

interfejsy

służą do przechowywania funkcji i procedur jednakże nie zawierają części implementacyjnej.

ciało

atributy i zaimplementowane metody.

Języki i metodyka programowania

Język C# obiekty i klasy

```
public class Figura{  
    double a,b;  
    double pole(){  
        return a*b;  
    }  
}
```

Języki i metodyka programowania

Język C# obiekty i klasy

Konstruktor to metoda zdefiniowana w danej klasie.

- Nazwa konstruktora jest taka sama jak nazwa klasy.
- Dostęp do konstruktora jest public
- Konstruktor może mieć parametry.
- Konstruktor może być przeciążany (nadpisywany).
- Konstruktor nie definiuje typu wartości zwracanej.
- Konstruktor jest wywoływany przy tworzeniu obiektu danej klasy za pomocą polecenia **new**.

```
public class Figura{  
    public Figura(double x,double y){  
        a=x;b=y;  
    }  
    double a,b;  
    ...  
}
```

Języki i metodyka programowania

Język C# obiekty i klasy

Destruktor to metoda zdefiniowana w danej klasie.

- Nazwa destruktora jest taka sama jak nazwa klasy poprzedzona ~.
- Destruktor nie definiuje typu wartości zwracanej.
- Destruktor jest wywoływany przy usuwaniu obiektu danej klasy.

```
public class Figura{  
    public Figura(double x,double y){  
        a=x;b=y;  
    }  
    ~Figura(){  
    }  
    double a,b;  
    ...  
}
```

Języki i metodyka programowania

Język C# obiekty i klasy

Pierwszy program:

C:

```
#include <stdio.h>
int main()
{
    printf("Hello world!");
}
```

C#:

```
using System;
public class HelloWorld
{
    public static int Main()
    {
        Console.WriteLine("Hello world!");
    }
}
```