

Přetížení operátorů Delegáty a události

9. cvičení

Jiří Zacpal

KMI/ZP3CS – Základy programování 3 (C#)

Prezentace projektů

- Svoje řešení student prezentuje v posledních třech cvičeních. Za prezentaci může student získat **0 až 3 body**. Průběh prezentace (10 minut):
 - Předvedení fungujícího programu.
 - Ukázka klíčových částí kódu.

Přetížení operátorů

- použití operátorů pro jiné datové typy
- příklad:

```
class Hodina
{
    private int hodnota
    public Hodina(int ph){this.hodnota=ph;}

    public static Hodina operator + (Hodina ls, hodina ps)
    {return new Hodina(ls.hodnota+ps.hodnota);}
```

```
Hodina h1=new hodina(10),h2=new hodina(15),h3;
h3=h1+h2;
```

Pravidla při přetížení

- operátor je veřejný
- operátor je statický
- operátor není nikdy polymorfní
- u operátoru nelze použít slova virtual, abstract, override nebo sealed
- nelze měnit aritu, prioritu a asociativitu operátorů
- nelze vytvářet nové operátory
- nelze měnit význam operátorů pro vestavěné typy
- některé operátory nelze přetěžovat (například .)

Symetrické operátory

- příklad:

```
class Hodina
{
    private int hodnota
    public Hodina(int ph){this.hodnota=ph;}

    public static Hodina operator + (Hodina ls, hodina ps)
    { return new Hodina(ls.hodnota+ps.hodnota);}

    public static Hodina operator + (Hodina ls, int ps)
    {return ls + new Hodina(ps);}

    public static Hodina operator + (int ls, hodina ps)
    {return new Hodina(ls)+ ps;}

}
```

Přetěžování operátorů uspořádání

- je nutné přetěžovat vždy oba operátory:
 - == a !=
 - < a >
 - <= a >=

Převodní operátory

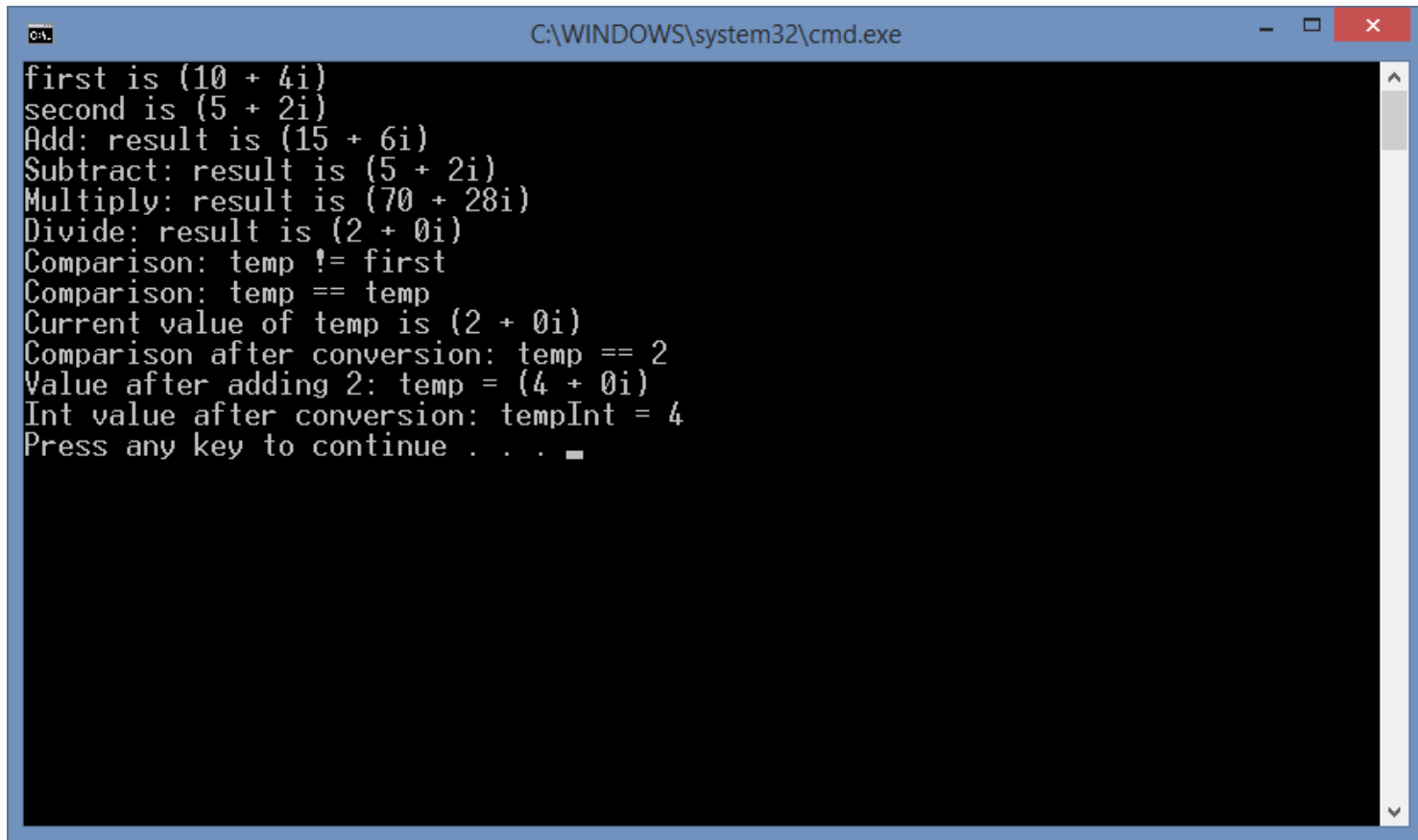
- umožňuje definovat implicitní nebo explicitní možnost převodu mezi datovými typy

- příklad:

```
class Hodina
{
    ...
    public static implicit operator int (Hodina arg)
    {return arg.hodnota}
    nebo
    public static explicit operator int (Hodina arg)
    {return arg.hodnota}

}
Hodina h(10);
int i=h;
int i=(int)h;
```

Příklad 1



A screenshot of a Windows command prompt window titled "C:\WINDOWS\system32\cmd.exe". The window has a blue title bar with standard Windows window controls (minimize, maximize, close). The command prompt shows the output of a C# program. The text is as follows:

```
first is (10 + 4i)
second is (5 + 2i)
Add: result is (15 + 6i)
Subtract: result is (5 + 2i)
Multiply: result is (70 + 28i)
Divide: result is (2 + 0i)
Comparison: temp != first
Comparison: temp == temp
Current value of temp is (2 + 0i)
Comparison after conversion: temp == 2
Value after adding 2: temp = (4 + 0i)
Int value after conversion: tempInt = 4
Press any key to continue . . .
```


Delegát

- delegát je ukazatel na metodu nebo více metod, kterou můžete zavolat prostřednictvím delegátu tak, že uvedete jeho název
- delegáty lze dynamicky měnit
- deklarace:

delegate typ nazevtypudelegatu(par);

- delegátový typ definuje tvar metody na kterou bude odkazovat

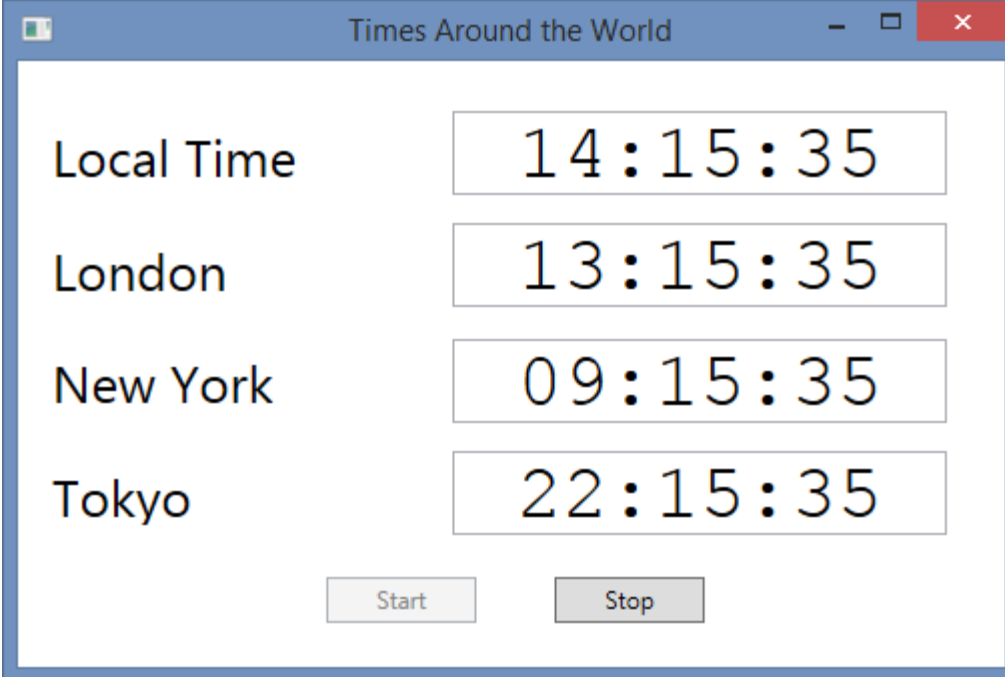
typ nazevtypudelegatu nazevdelegatu;

- konkrétní delegát

- připojení metody k delegátu:

nazevdelegatu+=metoda;

Příklad 2



The screenshot shows a Windows application window titled "Times Around the World". Inside the window, there are four rows of time information. Each row consists of a location name on the left and a time display on the right. The time displays are in a digital format showing hours, minutes, and seconds separated by colons. At the bottom of the window, there are two buttons: "Start" and "Stop".

Location	Time
Local Time	14:15:35
London	13:15:35
New York	09:15:35
Tokyo	22:15:35

Start Stop