

Proměnné, operátory a výrazy

2. cvičení

Jiří Zacpal

KMI/ZP3CS – Základy programování 3 (C#)

Základní pojmy

- Příkazy
- Metody - funkce
- Identifikátory
- Proměnná
 - deklarace proměnné
 - před použitím je nutné proměnné přiřadit hodnotu

Primitivní datové typy

- int 4 B
- long 8 B
- float 4 B
- double 8 B
- decimal 16 B (pro uložení peněžních hodnot)
- string 2 B na znak
- char 2 B
- bool 1 B (logická hodnota true nebo false)

Aritmetické operátory

- + , - , * , / , % (modulo)
- lze je použít s číselnými typy
- pro typ string lze použít + pro spojení řetězců
- typy operátorů
- vlastnosti:
 - asociativita
 - priorita
 - arita

Konverze mezi typy

`Convert.metoda(co);`

`typ.Parse(co);`

- umožňuje konvertovat hodnoty mezi datovými typy
- příklad:

`a = Convert.ToInt32(„82“);`

`a = int.Parse(„82“);`

Implicitně typované lokální proměnné

```
var identifikator = hodnota;
```

- typ proměnné je odvozen z typu inicializační hodnoty
- příklad:

```
var a = 8;
```

```
var b = „Jedna“;
```

Metody

- metoda = pojmenovaná posloupnost příkazů (podobně jako funkce)
- deklarace:

```
typ JmenoMetody(parametry)
{
    //tělo metody
    return NavratovaHodnota;
}
```

- volání:

```
vysledek=JmenoMetody(hodnoty
parametru)
```

Příklad 1

Aritmetické operátory

levý operand

10

pravý operand

8

☐ + sčítání

☒ - odčítání

☐ * násobení

☐ / dělení

☐ % zbytek

Vypočti

Výraz: 10 - 8

Výsledek: 2

Konec

Obor platnosti

- lokální obor platnosti
- třídní obor platnosti

Přetěžování metod

= jsou-li dva identifikátory stejné (ve stejném oboru platnosti)

- metody se musí lišit počtem nebo typem parametrů

Nepovinné parametry

- některé parametry jsou nepovinné
- definice:

```
typ Jmeno(PevnePar, typ vol1=h, typ  
vol2=h)
```

- příklad:

```
void volitelne(int prvni, double  
druhy=0.0, string tretí=„Ahoj“) {...};
```

```
volitelne(99,123.45,“Světě”);
```

```
volitelne(100,54.321);
```

```
volitelne(treti:“Světě“,prvni:100);
```

Přetížené metody s volitelnými parametry

- příklad:

```
void volitelne(int prvni, double druhy=0.0, string  
treti=„Ahoj“){...};
```

```
void volitelne(int prvni, double druhy=0.0, string  
treti=„Ahoj“, int ctvrty=100) {...};
```

```
volitelne(99,123.45,“Světě”);
```

1. metoda

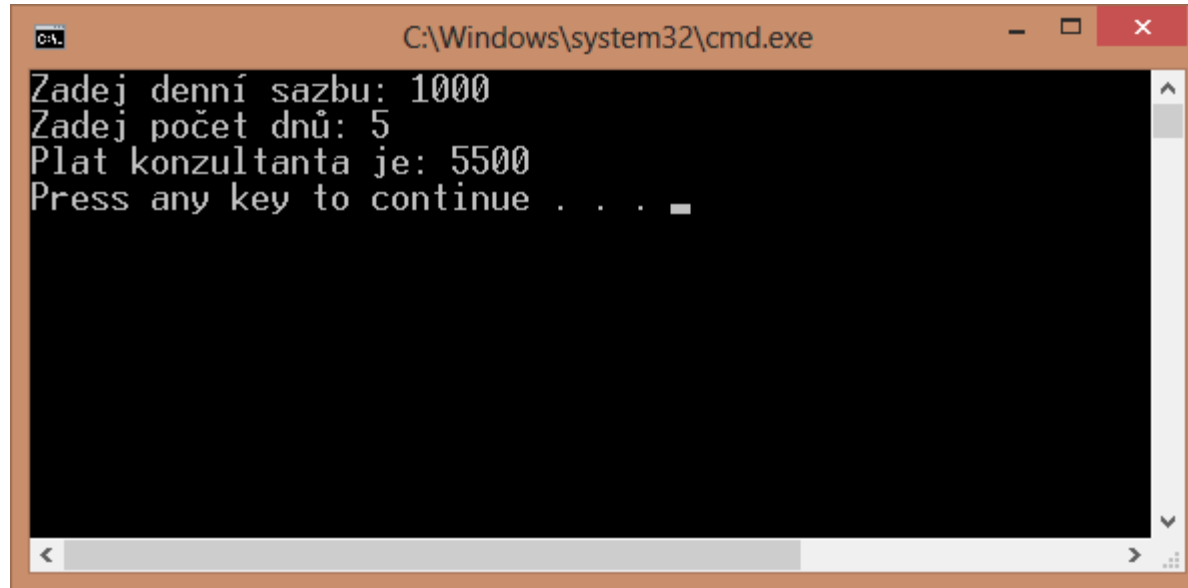
```
volitelne(100,ctvrty:54);
```

2. metoda

```
volitelne(1,2.5);
```

nejednoznačné volání metody

Příklad 2



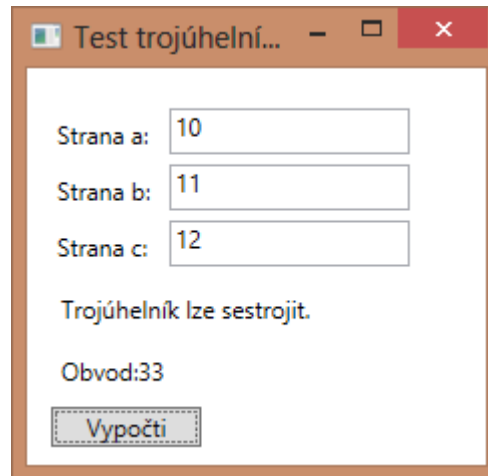
A screenshot of a Windows command prompt window. The title bar shows the path `C:\Windows\system32\cmd.exe`. The window contains the following text:

```
Zadej denní sazbu: 1000  
Zadej počet dnů: 5  
Plat konzultanta je: 5500  
Press any key to continue . . .
```

The text is displayed in a monospaced font on a black background. The window has a standard Windows border with minimize, maximize, and close buttons in the top right corner. A scrollbar is visible on the right side of the text area.

Úkol

- Vytvořte grafickou aplikaci, která ověří, jestli lze sestrojit trojúhelník se zadanými velikostmi stran. Současně aplikace také spočítá obvod trojúhelníku.



The screenshot shows a Windows application window with the title bar 'Test trojúhelní...'. The window contains three text input fields labeled 'Strana a:', 'Strana b:', and 'Strana c:'. The values entered are 10, 11, and 12 respectively. Below the inputs, the text 'Trojúhelník lze sestrojit.' is displayed. Underneath that, the text 'Obvod:33' is shown. At the bottom, there is a button labeled 'Vypočti'.