

Číslo a název šablony	III/2 Inovace a zkvalitnění výuky prostřednictvím ICT
Číslo didaktického materiálu	EU-OPVK-VT-III/2-ŠR-116
Druh didaktického materiálu	DUM
Autor	RNDr. Václava Šrůtková
Jazyk	čeština
Téma sady didaktických materiálů	Programování v C# v příkladech I
Téma didaktického materiálu	Složené přiřazení, členské a lokální proměnné
Vyučovací předmět	Seminář z informatiky
Cílová skupina (ročník)	Žáci ve věku 16–17 let
Úroveň žáků	začátečníci
Časový rozsah	1–2 vyučovací hodiny
Klíčová slova	Složené přiřazení, členské a lokální proměnné, kompatibilita vzhledem k přiřazení
Anotace	Studenti se učí pracovat se složeným přiřazovacím příkazem, rozlišovat vhodnost použití členských a lokálních proměnných
Použité zdroje	<p>DRÓZD, Januš a Rudolf KRYL. <i>Začínáme s programováním</i>. 1.vyd. Praha: Grada, 1992, 306 s. ISBN 80-854-2441-X.</p> <p>ELLER, Frank. <i>C# - začínáme programovat: podrobný průvodce začínajícího uživatele</i>. 1. vyd. Praha: Grada, 2002, 240 s. ISBN 80-247-0324-6.</p> <p>VYSTAVĚL, Radek. <i>Moderní programování: sbírka úloh k učebnici pro začátečníky</i>. 2. vyd. Ondřejov: moderníProgramování, 2008, 2 sv. ISBN 978-80-903951-5-2.</p> <p>VYSTAVĚL, Radek. <i>Moderní programování: učebnice pro začátečníky</i>. Ondřejov: moderníProgramování s.r.o, 2007, 2 sv. ISBN 978-80-903951-0-7.</p>
Typy k metodickému postupu učitele, doporučené výukové metody, způsob hodnocení, typy k individualizované výuce apod.	<p>Text je možno využít ke společné práci, samostatné přípravě studentů, domácímu studiu apod.</p> <p>Při společné práci je vhodné nejprve obtížnější úlohy rozebrat, potom společně se studenty implementovat na počítači. (Rozbor nejlépe na tabuli, synchronní řešení s promítáním)</p> <p>Prezentace obsahuje stručné shrnutí poznatků potřebných pro řešení příkladů. V pracovním listu je zadání cvičení – většinou se jedná o úlohy, které by měli studenti naprogramovat samostatně. Není nutné, aby všichni zpracovali všechno, vhodné je diferencovat podle jejich</p>

	<p>zájmu a schopností. Obtížnější úlohy jsou označeny hvězdičkou. Součástí materiálu je zdrojový kód těchto příkladů.</p> <p>Návrh způsobu hodnocení: ohodnocení samostatné práce během hodiny např. podle volby a počtu úloh a elaborace řešení (efektivnost, komentáře...).</p>
--	---

Metodický list k didaktickému materiálu

Prohlášení autora

Tento materiál je originálním autorským dílem. K vytvoření tohoto didaktického materiálu nebyly použity žádné externí zdroje s výjimkou zdrojů citovaných v metodickém listu.

Obrázky (schémata a snímky obrazovek) pocházejí od autora.

116. Složené přiřazení, členské a lokální proměnné, náhodná čísla

Složené přiřazení

Příklad 1

Na formulář umístíme dvě textová políčka, jedno z nich – `textBoxVstup` – bude sloužit k zadávání slov, která budeme chtít postupně přidávat do druhého políčka `textBoxZapis`. Pro druhé políčko nastavíme vlastnost **MultiLine** a **ReadOnly** na **True**, **Scrollbars** na **Both** (obojí rolóvátka). Políčko po přemístění na formulář roztáhneme myší.

Tlačítko **ButtonPridej** bude sloužit k přesouvání textu z prvního `textBoxu` do druhého – při jeho stisknutí je tedy třeba k textu v políčku `textBoxZapis` přidat odřádkování a text z prvního políčka.

Přidání hodnoty se dá realizovat dvěma způsoby:

proměnná = proměnná + hodnota;

Tedy: vyhodnoť pravou stranu – vezmi obsah proměnné a přičti k ní hodnotu a vlož výsledek do strany levé. V C# se dá akce zapsat jednodušeji (a možná i srozumitelněji)

příkazem složeného přiřazení: **proměnná += hodnota;**

Pro symbol konce řádku se používá statická vlastnost třídy **Environment.NewLine**.

```
private void buttonPridej_Click(object sender, EventArgs e)
```

```
{
```

```

        textBoxZapis.Text += textBoxVstup.Text + Environment.NewLine;

        textBoxVstup.Text = null;

        textBoxVstup.Focus();
    }

```

Příklad 2

Na formulář umístíme dvě tlačítka – s textem +1 a -1 a textové pole textBoxCislo, do kterého vložíme na začátek třeba číslo 15. Stisknutím tlačítek se bude velikost celého čísla adekvátně měnit.

```

private void buttonPlus_Click(object sender, EventArgs e)
{
    int x = Convert.ToInt32(textBoxCislo.Text);

    x += 1;

    textBoxCislo.Text = Convert.ToString(x);
}

private void buttonMinus_Click(object sender, EventArgs e)
{
    int x = Convert.ToInt32(textBoxCislo.Text);

    x -= 1;

    textBoxCislo.Text = Convert.ToString(x);
}

```

Protože se přidávání nebo ubírání jedničky často používá, můžete místo příkazu `x += 1` psát `x ++`. (stejně pro mínus 1 --) **++ se nazývá operátor inkrementace, -- operátor dekrementace.**

Složené přiřazení se používá také pro další aritmetické operátory jako např. násobení a dělení. (`x *= p` vynásobí obsah proměnné x číslem p)

Příklad 3

Co kdybychom chtěli počítat, kolikrát bylo kliknuto na tlačítko?

Přidáme tlačítko buttonKlik a budeme v něm při každém kliknutí zvyšovat hodnotu celočíselné proměnné, kterou při vytvoření formuláře vynulujeme:

```

private void Form1_Load(object sender, EventArgs e)
{

```

```

        int pocet = 0;
    }

    private void buttonKlik_Click(object sender, EventArgs e)
    {
        pocet++;

        textBoxPocetkliku=Convert.ToString(pocet);
    }

```

Jenže ono to nejde.

Problém je v tom, že existence proměnné končí na složené závorce, která uzavírá blok, v němž je tato proměnná deklarována. V našem příkladu potřebujeme v několika metodách používat tutéž proměnnou – musí být tedy deklarována někde jinde.

Třída okna programu

je vytvářena designerem pro náš program:

```

public partial class Form1 : Form
{
    public Form1()
    {
        InitializeComponent();
    }
}

```

Slovo **class** zahajuje definice nové třídy **Form1**, která je odvozena od třídy **Form**. Má všechny její složky, (Vlastnosti Text, BackColor, události Load, Paint, metody Close, Refresh) ale navíc právě metody, které my programujeme. (buttonMinus_Click) a také členské proměnné. (dále)

Členské proměnné

jsou proměnné definované ve třídě – v našem případě ve třídě okna. Proměnným, které jsme užívali až dosud a definovaly se v metodách se říká **lokální**. Právě taková musí být proměnná počet v příkladě 3. Pro přehlednost se členské proměnné zapisují na začátek definice třídy:

Dokončení příkladu 3:

```

public partial class Form1 : Form
{
    int pocet=0;

    public Form1()

```

...

```

private void buttonKlik_Click(object sender, EventArgs e)
{
    pocet++;

    textBoxPocetkliku.Text=Convert.ToString(pocet);
}

```

Náhodná čísla

Generování náhodných čísel se používá při hrách, simulacích... (Vzpomeňte na Excel). V zásadě se vždy vyjde z jednoho čísla, ze kterého se podle určitého algoritmu vypočítá další. (to dělá počítač) Aby šlo o čísla náhodná (říká se jim **pseudonáhodná**), musí mít každé stejnou pravděpodobnost, že bude generováno a nesmí být uhodnutelné předem.

Pro vytvoření náhodného čísla je třeba zajistit následující dva kroky:

Random nahoda = new Random() – vytvoření instance třídy Random

int cislo = nahoda.Next(min, max) – náhodné číslo vzniká metodou Next této instance, může mít hodnotu od min do max – 1.

Příklad 4

Zkusíme modelovat házení kostkou – náhodná čísla po stisknutí tlačítka vždy přidáme do víceřádkového textBoxuKostka.

```

public partial class Form1 : Form
{
    Random nahoda=new Random();

    public Form1()
    ...

private void buttonNahoda_Click(object sender, EventArgs e)
{
    int hod = nahoda.Next(1, 7);

    textBoxKostka.Text += Convert.ToString(hod)+Environment.NewLine;
}

```

Co kdybychom chtěli vypočítat, kolikrát padla šestka? Pak potřebujeme počítadlo šestek – celočíselnou členskou proměnnou.

```

public partial class Form1 : Form
{

```

```

        int pocet6 = 0;

...

private void buttonNahoda_Click(object sender, EventArgs e)
{
    int hod = nahoda.Next(1, 7);

    if (hod == 6)
        pocet6++;

    textBoxKostka.Text += Convert.ToString(hod)+Environment.NewLine;
}

private void buttonP6_Click(object sender, EventArgs e)
{
    MessageBox.Show("Šestka padla " + Convert.ToString(pocet6) +
"krát.");
}

```

Přehled a kombinace číselných typů

Víme už, že pro celá čísla užíváme typ **int**, pro desetinná **double**. Z historických důvodů existují jejich další podtypy, lišící se rozsahem hodnot, který je dán počtem bytů, která čísla zabírají v paměti.

Například celé číslo typu **byte**, které zabírá 8 bitů, může nabývat hodnot 0 až 255.

Náš známý typ **int** má rozsah přibližně ± 2 miliardy a ukládá se na 4 B.

U desetinných čísel navíc rozsah souvisí s přesností – **double** umožňuje přesnost 15 míst, zobrazuje $\pm 10^{308}$ hodnot a ukládá se do 8 B.

Přesné údaje si můžete vyhledat v dokumentaci, zde uvádíme pouze přehled.

Celá čísla: byte, sbyte, ushort, short, uint, int, ulong, long

Desetinná čísla: float, double

Desetinná čísla s přesnou reprezentací: decimal

Kombinace číselných typů – kompatibilita vzhledem k přiřazení

Int do double přiřadit jde, naopak nikoliv. Dá se provést typová konverze, prostudujte si následující příklad::

```
int y = 2;
```

double x = y; (jde implicitně)

int pocet = (int) x/3; (musí se explicitně přetypovat)

Důležité

Složené přiřazení a + = b přidej do a b, lze užít i další aritmetické operátory

Inkrementace a + +

Přidávání nového řádku v textu Environment.NewLine

Členské proměnné deklarují se ve třídě formuláře
`public partial class Form1 : Form`
`{TADY`

Náhodná čísla **Random nahoda = new Random()** – vytvoření instance třídy Random

int cislo = nahoda.Next(min, max) – náhodné číslo vniká metodou Next této instance, může mít hodnotu od min do max – 1.

Pracovní list

Cvičení

1. Modelujte hod mincí – náhodné číslo 0 je hlava, 1 je orel. Zobrazte výsledky hodu do víceřádkového textBoxu a vyhodnoťte, kolikrát co padlo.

2. (*) Napište program na zkoušení násobilky: Při stisknutí tlačítka Zadej příklad se dvě náhodná čísla uloží do textBoxů (textBox1, textBox2), do textBox3 uloží výsledek uživatele. Při stisknutí tlačítka Kontrola se zobrazí zpráva, bylo-li řešení správné. Při stisknutí tlačítka Hodnocení se zobrazí, kolik bylo příkladů celkem, kolik z nich správně. Můžete také provést oznámkování – nad 90% správně jednička, nad 80% – 90% dvojka,... pod 50% pětka.

K oběma příkladům můžete přidat tlačítko, které vynuluje členské proměnné, aby se dalo zase začít od začátku.

Řešení

`namespace Cvičení`

```
{  
  
    public partial class Form1 : Form  
  
    {  
  
        Random nahoda=new Random();  
  
        int pocet0=0;  
  
        int pocet1=0;  
  
        int otazky = 0;
```

```
int otazkyOK = 0;

int x, y; //činitelé pro násobení

public Form1()
{
    InitializeComponent();
}

private void buttonHod_Click(object sender, EventArgs e)
{
    //hod mincí

    int hod = nahoda.Next(0, 2);

    textBoxHody.Text += Convert.ToInt32(hod) + Environment.NewLine;

    if (hod == 0)
        pocet0++;

    else
        pocet1++;

}

private void buttonVysledek_Click(object sender, EventArgs e)
{
    MessageBox.Show("Hlava: " + Convert.ToString(pocet0) +
        Environment.NewLine + "Orel: " + Convert.ToString(pocet1));

}

private void buttonVynuluj_Click(object sender, EventArgs e)
{
    pocet0 = 0;

    pocet1 = 0;

    otazky = 0;

    otazkyOK = 0;

}
```



```

private void buttonPriklad_Click(object sender, EventArgs e)
{
    //zadání příkladu
    x = nahoda.Next(0, 11);
    y = nahoda.Next(0, 11);
    textBox1.Text = Convert.ToString(x);
    textBox2.Text = Convert.ToString(y);
    textBox3.Text = null;
    otazky++;
    textBox3.Focus();
}

```

```

private void buttonKontrola_Click(object sender, EventArgs e)
{
    //kontrola uživatelského řešení
    int z = x * y;
    try
    {
        if (z == Convert.ToInt32(textBox3.Text))
        {
            otazkyOK++;
            MessageBox.Show("správně");
        }
        else
            MessageBox.Show("chyba, má být " + z.ToString());
    }
    catch
    {
        MessageBox.Show("Zadejte čísla! ");
    }
}

```

```

        textBox3.Text = null;

        textBox1.Text = null;

        textBox2.Text = null;

    }

    private void buttonZnamka_Click(object sender, EventArgs e)
    {
        //hodnocení známkou

        double procent=(double)otazkyOK/otazky*100;

        //když se tady neprovede explicitní konverze, je dělení celočíselné

        int znamka;

        if (procent>90)znamka=1;

        else

            if(procent>80)znamka=2;

            else

                if(procent>70)znamka=3;

                else

                    if(procent>60)znamka=4;

                    else znamka=5;

        MessageBox.Show("Otázek celkem: " + Convert.ToString(otazky) +
            Environment.NewLine + "Otázek správně: " + Convert.ToString(otazkyOK) +
            Environment.NewLine + "Známka celkem: " + Convert.ToString(znamka));
    }
}
}

```