

Číslo a název šablony	III/2 Inovace a zkvalitnění výuky prostřednictvím ICT
Číslo didaktického materiálu	EU-OPVK-VT-III/2-ŠR-216
Druh didaktického materiálu	DUM
Autor	RNDr. Václava Šrůtková
Jazyk	čeština
Téma sady didaktických materiálů	Programování v C# v příkladech II
Téma didaktického materiálu	Zpracování textového souboru
Vyučovací předmět	Seminář z informatiky
Cílová skupina (ročník)	Žáci ve věku 17–18 let
Úroveň žáků	Mírně pokročilí
Časový rozsah	1–2 vyučovací hodiny
Klíčová slova	Textový soubor, otevření, zavření
Anotace	Studenti programují otevření, zpracování a zavření textového souboru
Použité zdroje	<p>TÖPFEROVÁ, Dana a Pavel TÖPFER. <i>Sbírka úloh z programování</i>. Vyd. 1. Praha: Grada, 1992, 98 s. Educa '99. ISBN 80-854-2499-1.</p> <p>VYSTAVĚL, Radek. <i>Moderní programování: sbírka úloh k učebnici pro středně pokročilé</i>. 1. vyd. Ondřejov: moderníProgramování, 2008-2009, 2 sv. ISBN 978-80-903951-3-8.</p> <p>VYSTAVĚL, Radek. <i>Moderní programování: učebnice pro středně pokročilé</i>. Ondřejov: moderníProgramování s.r.o, 2008. ISBN 978-80-903951-2-1.</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 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</p>

	řešení (efektivnost, komentáře...).
--	-------------------------------------

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.

216. Zpracování textového souboru

Čtení textového souboru je podobné zápisu. Používáme instanci třídy **StreamReader**, která také patří do jmenného prostoru **System.IO** a její metodu **ReadLine()** pro čtení řádku. Instance StreamReader má buď jeden parametr – jméno souboru, (s kódováním UTF 8) nebo dva – druhým je kódování. Můžeme také pracovat s dialogem pro otevření souboru – **OpenFileDialog**.

Příklad 1

Otevřeme a přečteme po řádcích náš text se žluťoučkým koněm.

```
private void buttonOtevri_Click(object sender, EventArgs e)
{
    //Přečtení a zobrazení prvních dvou řádků TS
    string jmeno=null;//jmeno souboru
    string text = null;//obsah textového souboru
    DialogResult odp = openFileDialog.ShowDialog();
    if (odp == DialogResult.OK)
        jmeno=openFileDialog.FileName;
    //otevření souboru dialogem
    StreamReader soubor = new StreamReader(jmeno, Encoding.Default);
    //otevření souboru, Encoding.default je Windows CP 1250
    text += soubor.ReadLine();//1. řádek;
    MessageBox.Show(text);
    text += Environment.NewLine+soubor.ReadLine();//2. řádek;
    textBoxZobraz.Text = text;
    soubor.Close();//zavření souboru
}
```

Příklad 2.

Zobrazení souboru

Každá práce se souborem má tři části: otevření, zpracování, zavření.

Textové soubory se zpracovávají po řádcích a sekvenčně – na rozdíl od binárních, chceme-li nějaký údaj, musíme přečíst všechno před ním. Protože většinou dopředu nevíme, kolik řádků má soubor, využíváme toho, že na konci souboru vrací `ReadLine()` null a budeme ho číst cyklem `while`.

Vstupní textový soubor si nejprve připravte, nejlépe v Poznámkovém bloku.

```
string radek = null;

...

StreamReader soubor = new StreamReader(jmeno, Encoding.Default);

while ((radek = soubor.ReadLine()) != null)
    //do proměnné řádek načteme řádek z TS
    //a testujeme, nejsme-li na konci
    text += radek + Environment.NewLine;

textBoxZobraz.Text = text;
```

...

Příklad 3

Zobrazení, úprava – převod na velká písmena a následné uložení do nového souboru.

Vždy po přečtení řádku ze vstupního souboru ho převedeme na velká písmena a uložíme do souboru výstupního.

...

```
StreamReader soubor = new StreamReader(jmeno, Encoding.Default);

StreamWriter vystup = new StreamWriter("Velká písmena", false,
Encoding.Default);

while ((radek = soubor.ReadLine()) != null)
    //do proměnné řádek načteme řádek z TS
    //a testujeme, nejsme-li na konci
    {
        string radek2=radek.ToUpper();//metoda třídy string převedení
```

```

// na velká písmena
        vystup.WriteLine(radek2);
    }
    soubor.Close();//zavření souboru
    vystup.Close();
...

```

Příklad 4

Zobrazení, úprava a následné uložení téhož souboru – na konec každého řádku přidáme jeho délku.

I v tomto případě budeme pracovat se dvěma soubory, ale jeden z nich – dočasně výstupní – umístíme do dočasného adresáře, po skončení práce vstupní soubor smažeme a výstupní přesuneme na jeho místo.

```

...
{
    //Připsání počtu znaků na konec řádku, práce s dočasným adresářem
    string jmeno = null;
    string radek = null;
    string docasAdr = Path.GetTempPath();
    string nahodJmeno = Path.GetRandomFileName();
    string docasSoubor = Path.Combine(docasAdr, nahodJmeno);
    //vytvoření jména dočasného souboru necháme na systému
    DialogResult odp = openFileDialog.ShowDialog();
    if (odp == DialogResult.OK)
        jmeno = openFileDialog.FileName;
    StreamReader soubor = new StreamReader(jmeno, Encoding.Default);
    StreamWriter vystup = new StreamWriter(docasSoubor, false,
Encoding.Default);
    while ((radek = soubor.ReadLine()) != null)
    {
        int delka = radek.Length;
        string radek2 = radek + " "+delka.ToString();
        vystup.WriteLine(radek2);
    }
}

```

```

    }

    soubor.Close();//zavření souboru

    vystup.Close();

    File.Delete(jmeno);//smazání původního souboru

    File.Move(docasSoubor, jmeno);//přesunutí původního souboru na jeho
místo
}

```

Poznámka: více o metodách tříd Path, případně File se dočtete v dokumentaci C#.

Důležité

Čtení textového souboru – instance **StreamReader**

ReadLine() pro čtení řádku

OpenFileDialog – dialog pro otevření souboru.

Práce s adresáři – třída **Path**

Práce se soubory – třída **File**

Pracovní list

Cvičení

1. Upravte některý z předchozích či následujících příkladů, s použitím SaveFileDialogu.
- 2.(*). Zjistěte délku nejdelšího řádku vstupního souboru a zobrazte tento řádek.
3. Upravte vstupní soubor tak, že očísľujete jeho řádky. (Můžete použít buď práci s dočasným souborem, nebo vytvořit k vstupnímu souboru soubor výstupní s očísľovanými řádky.)

Řešení

1.

```

private void buttonUprava_Click(object sender, EventArgs e)
{
    //úprava soboru s oknem pro výběr vstupu i výstupu
    string vstup=null;
    string vystup=null;
    string radek=null;
    DialogResult odp = openFileDialog.ShowDialog();
    if (odp == DialogResult.OK)

```

```

        vstup = openFileDialog.FileName;
        DialogResult odp2 = saveFileDialog.ShowDialog();
        if (odp2 == DialogResult.OK)
            vystup = saveFileDialog.FileName;
        StreamReader soubor1 = new StreamReader(vstup, Encoding.Default);
        StreamWriter soubor2 = new StreamWriter(vystup, false,
        Encoding.Default);

        while ((radek = soubor1.ReadLine()) != null)
        {
            string radek2 = radek.ToUpper();
            soubor2.WriteLine(radek2);
        }
        soubor1.Close();
        soubor2.Close();
    }
}

```

2.

```

{
    //Nejdelší řádek
    string jmeno = null;
    string radek = null;
    string nejdelsi = null; //Nejdelší řádek
    int maxdelka = 0; //Jeho délka - abychom nemuseli soubor procházet
    //dvakrát, budeme si pamatovat oba údaje. (Stačil by i řádek)
    DialogResult odp = openFileDialog.ShowDialog();
    if (odp == DialogResult.OK)
        jmeno = openFileDialog.FileName;
    StreamReader soubor = new StreamReader(jmeno, Encoding.Default);
    while ((radek = soubor.ReadLine()) != null)
    {
        int delka = radek.Length;
        if (delka > maxdelka)
        {

```

```

        maxdelka = delka;

        nejdelsi = radek;

    }

}

soubor.Close();

MessageBox.Show(nejdelsi + Environment.NewLine +
    "Délka: " + maxdelka.ToString());

}

```

3.

```

{
    //očíslování řádků vstupního souboru

    string vstup = null;

    string vystup = "Blabla.Txt";

    string radek = null;

    int cislo = 1; //čísla řádků od 1

    //úprava souboru s oknem pro výběr vstupu i výstupu

    DialogResult odp = openFileDialog.ShowDialog();

    if (odp == DialogResult.OK)

        vstup = openFileDialog.FileName;

    StreamReader soubor1 = new StreamReader(vstup, Encoding.Default);

    StreamWriter soubor2 = new StreamWriter(vystup, false,
        Encoding.Default);

    while ((radek = soubor1.ReadLine()) != null)

    {

        string radek2 = cislo.ToString()+" "+radek;

        soubor2.WriteLine(radek2);

        cislo++;

    }

    soubor1.Close() soubor2.Close();

    File.Delete(vstup);

    File.Move(vystup, vstup);

}

```

