

Číslo a název šablony	III/2 Inovace a zkvalitnění výuky prostřednictvím ICT
Číslo didaktického materiálu	EU-OPVK-VT-III/2-ŠR-213
Druh didaktického materiálu	DUM
Autor	RNDr. Václava Šrůtková
Jazyk	čeština
Téma sady didaktických materiálů	Programování v C# v příkladech II
Téma didaktického materiálu	Vnořené cykly
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	Cykly for, while, prvočíslo
Anotace	Studenti se učí programovat vnořené cykly na příkladech z grafiky a na vlastnostech čísel.
Použité zdroje	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. 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. 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. VYSTAVĚL, Radek. <i>Moderní programování: učebnice pro středně pokročilé</i>. Ondřejov: moderníProgramování s.r.o, 2008. ISBN 978-80-903951-2-1. VYSTAVĚL, Radek. <i>Moderní programování: učebnice pro začátečníky</i>. Ondřejov: moderníProgramování s.r.o, 2007, 2 sv. ISBN 978-80-903951-0-7.
Typy k metodickému postupu učitele, doporučené výukové metody, způsob hodnocení, typy k individualizované výuce apod.	Při společné práci je vhodné nejprve obtížnější úlohy rozebrat, potom společně se studenty implementovat na počítači. (Rozbor nejlépe na tabuli, synchronní řešení s promítáním) 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. Návrh způsobu hodnocení: ohodnocení samostatné práce během hodiny např. podle volby a počtu úloh a elaborace 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.

213. Vnořené cykly

Stejně jako podmíněné příkazy, také cykly lze do sebe vnořovat. Použití si ukážeme nejprve na příkladech, které se dají efektivněji řešit s využitím vlastních metod, jejichž programování se budeme brzy věnovat více – a potom na typických úlohách – zpracování dvourozměrných objektů jako jsou obrázky, tabulky apod.

Příklad 1.

Vypište všechna prvočísla menší než 100.

Budeme procházet přirozená čísla od dvou do sta a u každého zjišťovat, zda se jedná o prvočíslo. Test prvočíselnosti přitom funguje tak, že zkoumáme postupně čísla od dvou do odmocniny z daného a hledáme dělitele. Pokud ho nenajdeme, je číslo prvočíslo.

Vnější cyklus tedy zahrnuje procházení čísel od 2 do 100, vnitřní zjišťování prvočíselnosti.

```
private void buttonPrvoci_Click(object sender, EventArgs e)
{
    //výpis prvočísel od 2 do 100
    for (int cislo = 2; cislo < 101; cislo++)
    {
        //vnější cyklus - kolikáté číslo testujeme
        {
            int del = 2; //1. adept na dělitele
            int odm = Convert.ToInt32(Math.Sqrt(cislo));
```

```

        while ((cislo % del != 0) && (del <= odm))

            del++;

        //vnitřní cyklus - zjišťování, zda cislo je prvočíslo

        if (del > odm)

            textBoxVypis.Text += cislo.ToString()+Environment.NewLine;

        //výpis prvočísla

    }

}

```

Příklad 2.

Naplňte panel obdélníkem z koleček.

```

private void panelCtver_Paint(object sender, PaintEventArgs e)

{

    Graphics kp=e.Graphics;

    int ps = panelCtver.Width;

    int pr = panelCtver.Height;

    if (radioButtonKol.Checked)

        for (int x = 20; x < ps-16; x+=20)

            //kreslíme sloupec na x-ovou pozici

            for (int y = 20; y < pr-16; y+=20)//pr

                //kreslíme kolečka ve sloupci

                kp.DrawEllipse(Pens.Blue, x - 8, y - 8, 16, 16);

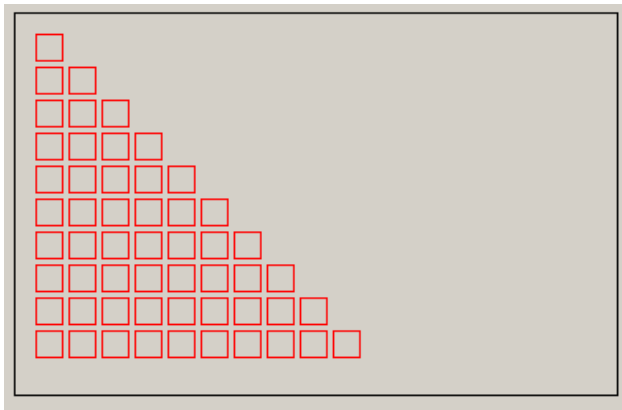
}

```

Pracovní list

Cvičení

1. Nakreslete na panel trojúhelník ze čtverečků podle obrázku.



2. Vypište dělitele čísel od 1 do 100
3. (*)Vypište prvních n prvočísel, kde n zadá uživatel
4. Zkuste si vymyslet a naprogramovat kreslení nějakého dalšího obrazce z koleček či čtverečků.

Řešení

1. ...

```
if (radioButtonObd.Checked)
    for (int x = 20; x <= pr-16; x += 20)
        //kreslíme x-tý sloupec,
        for (int y =x; y <=pr-16; y += 20)
            //První je celý, druhý začíná o čtvereček níž...
            kp.DrawRectangle(Pens.Red, x - 8, y - 8, 16, 16);
```

2.

```
private void buttonDelitele_Click(object sender, EventArgs e)
{
    //Dělitelé čísel od 1 do 100
    for (int cislo = 1; cislo < 101; cislo++)
    {
        int n = cislo / 2; //dělitele hledáme do poloviny
        string delitele = cislo.ToString()+" : ";
        for (int d=1;d<=n;d++)
            if (cislo%d ==0) //dělitele ukládáme do pomocného řetězce
                delitele+=d.ToString()+" ";
        textBoxVypis.Text += delitele + Environment.NewLine;
    }
}
```

```
}
```

3.

```
private void buttonPrvoci2_Click(object sender, EventArgs e)
{
    //prvních n prvočísel
    int cislo = 1;
    int n = Convert.ToInt32(textBoxPocet.Text);
    int p = 0; //počítadlo prvočísel
    while(p < n)
    {
        cislo++; //testování dalšího čísla
        int del = 2; //1. adept na dělitele
        int odm = Convert.ToInt32(Math.Sqrt(cislo));
        while ((cislo % del != 0) && (del <= odm))
            del++;
        //vnitřní cyklus - zjišťování, zda cislo je prvočíslo
        if (del > odm)
        {
            p++;
            textBoxVypis.Text += cislo.ToString() +
Environment.NewLine;
            //výpis prvočísla a zvětšení počtu prvočísel
        }
    }
}
```

4. Individuální – pokud kreslíme po sloupcích, budeme v závislosti na vnějším cyklu měnit hranice odkud kam se kreslí sloupec.

Např.

```
private void panel1_Paint(object sender, PaintEventArgs e)
{
    //trojuhelník z kolecek
    Graphics kp = e.Graphics;
    int s = panel1.Width;
    int v = panel1.Height;
```

```

//kreslení sloupců
for (int x=20;x<s-20;x+=20)
//sloupec koleček
for (int y=v;y>v-x;y-=20)
    kp.DrawEllipse(Pens.DarkGoldenrod, x, y, 20, 20);
}

```

Nakreslí:

