

Číslo a název šablony	III/2 Inovace a zkvalitnění výuky prostřednictvím ICT
Číslo didaktického materiálu	EU-OPVK-VT-III/2-ŠR-202
Druh didaktického materiálu	DUM
Autor	RNDr. Václava Šrůtková
Jazyk	čeština
Téma sady didaktických materiálů	Programování v C# v příkladech II
Téma didaktického materiálu	Příklady na cyklus s podmínkou
Vyučovací předmět	Seminář z informatiky
Cílová skupina (ročník)	Žáci ve věku 16–17 let
Úroveň žáků	Mírně pokročilí
Časový rozsah	1–2 vyučovací hodiny
Klíčová slova	Cyklus while, do-while, dělitelnost, Euklidův algoritmus
Anotace	Studenti
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.	Text je možno využít ke společné práci, samostatné přípravě studentů, domácímu studiu 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.

202. Cykly s podmínkou – další příklady

Příklad 1.

Určete největšího společného dělitele vstupujících dvou čísel podle definice – největší z dělitelů čísla.

Nejprve otestujeme, které z čísel je menší – to by mohl být případný adept. Potom ho budeme postupně zmenšovat o jedničku, dokud jím nebudou obě čísla dělitelná. Pozor: podmínka setrvání v cyklu je tedy negace toho, že jsou obě dělitelná nsd – takže buď jím není dělitelné jedno, nebo druhé.

```
private void buttonND_Click(object sender, EventArgs e)
{
    int c1 = Convert.ToInt32(textBoxC1.Text);
    int c2 = Convert.ToInt32(textBoxC2.Text);
    int nsd;

    //první adept na nsd je menší z obou čísel, budeme ho postupně
    //zmenšovat o jedničku, dokud jím nebudou obě čísla dělitelná.

    if (c1 > c2)
        nsd = c2;
    else
```

```

        nsd = c1;

        while ((c1 % nsd != 0) || (c2 % nsd != 0))
        {
            nsd--;
        }

        MessageBox.Show("Největší společný dělitel je " + nsd.ToString());

```

Euklidův algoritmus

Největšího společného dělitele dvou čísel jsme v předchozím příkladu programovali na základě definice (největší číslo, které dělí daná čísla), v matematice jste ho také počítali pomocí prvočíselného rozkladu. V každém případě ale musíte používat celočíselné dělení a testovat velké množství čísel.

Euklidův algoritmus pro výpočet největšího společného dělitele dvou čísel vychází ze skutečnosti, že nsd dvou čísel je také nsd libovolného z nich a jejich rozdílu.

Tedy:

$Nsd(a, b) = nsd(a - b, b)$ když $a > b$;

$Nsd(a, b) = nsd(b - a, a)$ když $a < b$;

$Nsd(a, b) = a = b$, když $a = b$.

Konkrétně:

$Nsd(32, 24) = nsd(24, 8) = nsd(16, 8) = nsd(8, 8) = 8$.

```

private void buttonE_Click(object sender, EventArgs e)
{
    //Euklidův algoritmus

    int a = Convert.ToInt32(textBoxA.Text);

    int b = Convert.ToInt32(textBoxB.Text);

    while (a != b)
    {
        if (a > b)
            a -= b;
        else
            b -= a;
    }
}

```

```

        MessageBox.Show("Nsd je " + a.ToString());
    }
}

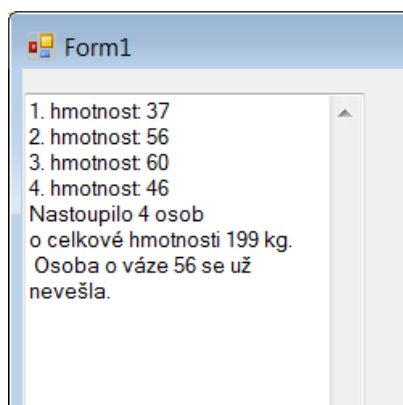
```

Pracovní list

Cvičení

1. (*)

Výtah má nosnost 500 kg. Nastupovat budou osoby o hmotnosti 10–100 kg. (Nástup modelujte generováním a zobrazením celého čísla v daném rozsahu.) Stisknutí tlačítka naplní výtah, aby jeho nosnost nebyla překročena zobrazí obsazení výtahu, případně informaci, kdo se už nevešel.



2. (*)

K danému přirozenému číslu najděte číslo, které má tytéž cifry v opačném pořadí.

3.

Ke dvěma daným přirozeným číslům nalezněte jejich nejmenší společný násobek – nejmenší číslo, které je dělitelné oběma vstupujícími čísly. Využijte této definice. (Jinak se dá zúročit známý vzorec: $\text{nsd}(a, b) \cdot \text{NSN}(a, b) = a \cdot b$.)

Řešení

1.

```

private void buttonVytah_Click(object sender, EventArgs e)
{
    textBoxVypisy.Text = null;
    int osoba=0;
    int i = 0;
    int celkem = 0;
    int nosnost = 200;
    while (celkem <= nosnost)
    {
        osoba = nahoda.Next(10, 101);
        celkem += osoba;
    }
}

```

```

        if (celkem <= nosnost)
        {
            i++;
            textBoxVypisy.Text += i.ToString() + ". " + "hmotnost: " +
osoba.ToString()+ Environment.NewLine;
        }
    }
    if (celkem > nosnost)
        celkem -= osoba;
    textBoxVypisy.Text += "Nastoupilo " + i.ToString() + " osob"+
        Environment.NewLine+"o celkové hmotnosti "
        + celkem.ToString() + " kg." + Environment.NewLine+
        " Osoba o váze " + osoba.ToString() + " se už nevešla.";
}

```

2.

```

private void button10brCif_Click(object sender, EventArgs e)
{
    int obrcif = 0; //Číslo s ciframi v opačném pořadí
    int cislo = Convert.ToInt32(textBoxCislo.Text);
    do
    {
        obrcif = obrcif*10+cislo % 10;
        cislo = cislo / 10;
    }
    while (cislo != 0);
    MessageBox.Show("Číslo s ciframi v opačném pořadí " + obrcif.ToString());
}

```

3.

```

private void buttonNSN_Click(object sender, EventArgs e)
{
    int c1 = Convert.ToInt32(textBoxC1.Text);
    int c2 = Convert.ToInt32(textBoxC2.Text);
    int nsn;

    //první adept je větší z obou čísel a budeme ho zvětšovat, dokud
    //nebude dělitelný oběma čísly - je zbytečné po jedné, lépe zkoušet
    //násobky většího čísla

    if (c1 > c2)

```

```
        nsn = c1;

else

        nsn = c2;

int vetsi=nsn;

while ((nsn % c1!= 0) || (nsn % c2!= 0))

{

        nsn+=vetsi;

}

MessageBox.Show("Nemenší společný násobek je " + nsn.ToString());

}
```