

Tematický celek 10

10.1 Cykly

Cykly slouží v procedurách událostí k provádění určité skupiny příkazů po pevný, předem daný počet opakování. Cyklus má proto smysl například při provádění několika spolu souvisejících výpočtů, při práci s elementy na obrazovce nebo při zpracování určité množiny vstupních dat.

10.1.1 Cyklus For ... Next

- použije se, pokud víme, kolikrát cyklus chceme provést.

```
For počítadlo = start To konec [Step přírůstek]
```

příkazy

CVIČNÝ 1

```
Next [počítadlo]
```

- provádění cyklu postupuje takto:
 1. Nastavení *počítadla* na hodnotu *start*.
 2. Otestuje se, zda je *počítadlo* větší než *konec*. Pokud ano, opustí se cyklus. (Je-li *přírůstek* záporný, testuje, je-li *počítadlo* menší než *konec*.)
 3. Provede *příkazy*.
 4. Zvětší hodnotu *počítadla* o 1 nebo o *přírůstek*, je-li nastaven.
 5. Opakuje kroky 2-4.

Novinkou ve Visual Basic (verze NET) je možnost deklarace proměnné přímo součástí cyklu pro zpřehlednění.

```
For a As Integer = 1 To 10 [Step přírůstek]
```

příkazy

```
Next [počítadlo]
```

10.1.2 Cykly Do...Loop

Do While

```
Do While podmínka
```

příkazy

CVIČNÝ 2

```
Loop
```

- použijeme tehdy, jestliže neznáme definitivní počet opakování cyklu.
- provádí se tehdy, jestliže je podmínka splněna.

Do Loop While

Do

příkazy

cvičný 3

Loop While *podmínka*

- cyklus se alespoň jednou provede.

Do Until

Do Until *podmínka*

příkazy

cvičný 4

Loop

- provádí se tehdy, jestliže podmínka není splněna.

Do Loop Until

Do

příkazy

cvičný 5

Loop Until *podmínka*

- cyklus se alespoň jednou provede.

10.2 InputBox

- můžeme použít tyto funkce, které umožňují vložit do aplikace předdefinovaná dialogová okna:
 - o InputBox – zobrazí výzvu a vrací cokoliv, co je uživatelem zadáno do textového pole.
 - o MsgBox – zobrazí zprávu a vrací hodnotu, určující, na které tlačítko uživatel klepl myší.

Funkce InputBox

- vybídne uživatele k zadání nějakých dat (textové, číselné hodnoty),
- tato funkce zobrazí modální dialogové okno, které vyžaduje od uživatele zadání údaje,
- zapisuje se takto:

InputBox (Prompt, Titul) As String

cvičný 6

- o *Prompt* – text, který se zobrazí v okně,
 - o *Titul* – nadpis okna.
- funkce vrací údaje zadané uživatelem ve formátu *String*.

10.3 Pole

- umožňuje odkazovat se na množinu proměnných,
- používá číslo – index k jejich odlišení,
- všechny prvky pole jsou stejného typu (není-li to pole hodnot typu Object).

Deklarace polí s pevnou délkou

- pole deklarujeme nastavením horní hranice v kulatých závorkách:

```
Dim Pole(18) As String
```

cvičný 7

Toto pole má 19 prvků s indexy 0-18,

- můžeme také definovat spodní hranici indexu:
`Dim Pole (1 To 10) As String`
Toto pole má indexy v rozmezí 1 až 10,

Vícerozměrná pole

- deklarace:
`Dim Pole(18,18) As String`
`Dim Pole(1 To 10, 2 To 5) As Integer`
`Dim Pole(18,10,5) As String`

Dynamická pole

- dynamické pole může být kdykoliv zvětšeno nebo zmenšeno,
- deklarace:
 - 1.Nadeklarujeme pole:
`Dim Pole()`
 - 2.Pomocí příkazu `ReDim` alokujeme aktuální počet prvků:
`ReDim Pole(18)`
- příkaz `ReDim` se může objevit jen v proceduře a používá stejnou syntaxi jako pole s pevnou velikostí (lze měnit dolní a horní hranici).

10.4 Metoda Print

Metoda **Print** a **PrintLine** představuje jistý speciální příkaz, který zobrazuje výstup do textového souboru nebo jej vytiskne na připojenou tiskárnu. Volání metody **Print** a **PrintLine** má následující syntaxi:

Print výraz

nebo pro zápis celého řádku

PrintLine výraz

cvičný 8

kde *výraz* je proměnná, vlastnost, textová hodnota nebo číselná hodnota (*výraz*) vypočtená v proceduře.

- pro vykreslování je optimální využít např. *textové pole*, *textový soubor* apod., z důvodu ochrany formuláře není možné metodu standardně používat pro ovládací prvky formulář a PictureBox,

FileOpen(1, "cesta k souboru", OpenMode.Output)

Otevře soubor pro čtení

PrintLine(1, "ted' píšu až na první řádek")

Zapíše informaci do textového souboru

FileClose(1)

Uzavře soubor

cvičný 8