

Studijní opora

Seminář 6

Ing. Michal Dostál, Ph.D.
katedra informatiky EF TUL

Obsah

6	Řídící struktury a cykly	2
6.1	Větvení kódu pomocí řídicích struktur	2
6.2	Cykly	3
6.2.1	Cyklus s končným počtem kroků	3
6.2.2	Cyklus s nekonečným počtem kroků	3
6.3	Úkoly k procvičení	4



Financováno
Evropskou unií
NextGenerationEU



Národní
plán
obnovy



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY

6 Řídící struktury a cykly

6.1 Větvení kódu pomocí řídicích struktur

Pokud potřebujeme v makru zajistit, aby se určitá část zdrojového kódu spustila pouze za určité podmínky, lze toho docílit pomocí tzv. podmíněného příkazu **If** nebo přepínače **Select**. Nejprve si ukážeme práci s příkazem **If** a souvisejícími příkazy **Else** a **End If**.

Základní tvar podmíněného příkazu je:

```
If podminka Then
    prikazy_ktere_se_provedou_pri_splneni_podminky
End If
```

V tomto základním tvaru program nejprve testuje, zda je splněna **podminka** a pokud ano, provede příkazy, které jsou za příkazem **Then** a před příkazy **End If**. Co může tvořit podmínku? Jakýkoliv výraz, který nabývá hodnot **True** nebo **False** a k tomu lze použít test rovnosti (pomocí **=**) nebo relační operátory jako **<**, **<=**, **>**, **>=**. Pokud potřebuje testovat výraz na nerovnost, použijeme zápis **<>**.

V případě, že potřebujeme program rozdělit do dvou větví, lze použít tento zápis:

```
If podminka Then
    prikazy_pri_splneni
Else
    prikazy_pri_nesplneni
End If
```

Taková zápis znamená, že pokud je splněna podmínka, provede se série příkazů, které jsou v první části, a to za **Then**. Pokud podmínka splněna není, provedou se příkazy uvedené v části začínající **Else**. Příklad takového zápisu může být:

```
moje_promenna = True
If moje_promenna = False Then
    MsgBox "proměnná je false!"
Else
    MsgBox "proměnná je true!"
End If
```

Tímto zápisem říkáme, že do proměnné **moje_promenna** je uložena hodnota **True**. Pomocí příkazu **If** pak testujeme, zda je v dané proměnné uložena hodnota **False**. Jelikož tato podmínka není splněna, provede se příkaz v sekci **Else**, a tedy vyskočí oznamovací okno s hláškou, že “proměnná je true!”.

Pokud potřebujeme program rozdělit na více než dvě větve, lze použít i příkaz **ElseIf**. Příklad takového kódu by pak vypadal následovně:

```
If moje_promenna = True Then
    MsgBox "Proměnná je True"
ElseIf moje_promenna = False Then
    MsgBox "Proměnná je False"
Else
    MsgBox "Proměnná nemá přiřazenou hodnotu"
End If
```

Dalším způsobem, jak rozdělit makro na více větví, je použití **Select**. Pokud se má zdrojový kód větvit v závislosti na obsahu jedné proměnné, je výhodnější (a přehlednější) použít příkaz **Select**. Příklad použití:

```
Select Case moje_promenna
    Case Is True
        MsgBox "Proměnná je True"
    Case Is False
        MsgBox "Proměnná je False"
    Case Else
        MsgBox "Proměnná nemá přiřazenou hodnotu"
End Select
```

Poslední větev obsahující **Case Else** se provede pouze v případě, že není splněna ani jedna z předchozích podmínek v jednotlivých větvích **Case**.

6.2 Cykly

Pokud je potřeba v makru provést nějaký příkaz opakovaně, lze k tomu použít cykly. Máme k dispozici dva typy cyklů: 1. cyklus s konečným počtem kroků 2. cyklus s nekonečným počtem kroků, kdy se při každém průběhu cyklu testuje, zda podmínka cyklu stále platí

6.2.1 Cyklus s končným počtem kroků

Představitelem cyklu s konečným počtem kroků je cyklus **For**, který v obecném tvaru vypadá následovně:

```
For promennna = pocatecni_hodnota To konecna_hodnota Step krok
    seznam_prikazu
Next promenna
```

Běžně pak cyklus vypadá například takto:

```
For i = 0 To 4
    MsgBox "Aktuální hodnota je " & i
Next i
```

Všimněte si, že příklad neobsahuje příkaz **Step**. Ten můžeme vynechat v případě, že krok cyklu je roven jedné.

6.2.2 Cyklus s nekonečným počtem kroků

Pro tento typ cyklu se používají dvě varianty příkazu **Do**, a to **Do While** a **Do Until**. Rozdíl je následující. U cyklu **Do While**, který má obecný zápis:

```
Do While podminka
    seznam_prikazu
Loop
```

seznam příkazů probíhá, dokud je splněná podmínka. Naopak u cyklu **Do Until** seznam příkazů probíhá, dokud nezačne podmínka platit. Zápis je prakticky stejný:

```
Do Until podminka  
    seznam_prikazu  
Loop
```

Pokud z nějakého důvodu potřebujeme předčasně ukončit cyklus, lze použít příkaz **Exit**.

6.3 Úkoly k procvičení

1. V podmínkové části příkazu **If** lze použít několik podmínek. Vyzkoušejte následující typy podmínek a popište, jaký je mezi nimi rozdíl:
 - `If podminka1 < 15 And (podminka2 = 43 Or podminka2 = 45) Then`
 - `If podminka1 < 15 And podminka2 = 43 Or podminka2 = 45 Then`
2. Vytvořte funkci, která do proměnné `urokova_sazba` uloží hodnotu úrokové sazby na základě následujících podmínek, týkajících se hodnoty proměnné `uspor` (použijte oba přístupy k větvení makra):
 - hodnota úspor na účtu je menší než 250 000 Kč: 5% úrok.
 - úspory jsou mezi 250 001 a 500 000 Kč: 4% úrok a
 - úspory jsou vyšší než 500 000 Kč: 3% úrok.
3. Pomocí cyklu vypište 3x hlášku “Ahoj světe” za použití `MsgBox`.