



## INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Projekt: 1.5, Registrační číslo: CZ.1.07/1.5.00/34.0304

### Funkce

Je standardní (zabudovaná v programovacím jazyku) jako např. SQRT nebo vytvořená uživatelem. Má svoje jméno a skládá se z jednoho nebo více příkazů. Pokud má funkce argumenty, jsou uvedeny v závorce za jménem funkce včetně svých typů. Voláním funkce dostaneme výsledek nazývaný funkční hodnota.

#### Příklad

Nadefinujme nyní funkci, která se bude jmenovat Trikrat a bude vracet trojnásobnou hodnotu vloženého čísla.



Ve zdrojovém kódu funkci deklarujeme za klíčovým slovem *implementation*:

```
var
  Form1: TForm1;
implementation
{$R *.DFM}
function Trikrat(a:Integer):Integer;
begin
  Trikrat:=a*3;
end;
```

Takto deklarovanou funkci můžeme pak používat na různých místech programu libovolně často. Obsluhu události *OnClick* tlačítka napíšeme tak, aby při stisknutí byla hodnota textu z editačního pole převedena na celé číslo a zapsána do proměnné **a**. Tato hodnota bude předána funkci Trikrat. Po výpočtu a převedení na řetězec bude výsledek zobrazen komponentou *Label2*. (Pro jednoduchost není program ošetřen proti chybnému zadání vstupní hodnoty.)

```
procedure TForm1.BitBtn1Click(Sender: TObject);
var A:Integer;
begin
```



## INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Projekt: 1.5, Registrační číslo: CZ.1.07/1.5.00/34.0304

```
a:=StrToInt(Edit1.Text);
Label2.Caption:=IntToStr(Trikrat(a));
end;
end
```

V deklaraci funkce je na posledním místě uveden typ výsledku – v našem případě Integer. Argument, který je uvedený v hlavičce funkce jako proměnná, se kterou bude funkce pracovat, je tzv. **formální parametr**. Jméno, které má, musí být použito v těle funkce. **Skutečný parametr**, který funkce při volání přebírá, může být proměnná jiného jména nebo také konstanta či výraz. Např.:

Trikrat(polomer)

Trikrat(2\*e-1)

Trikrat(4)

### Parametry volané hodnotou

Formálním parametrům se na začátku volání procedury přiřadí hodnota skutečného parametru. Změna proměnné uvnitř funkce neovlivní původní proměnnou. Pomocí parametrů volaných hodnotou se proceduře předávají vstupní hodnoty.

#### Příklad

Deklarujeme funkci, která hodnotu argumentu vynásobí 10. Do editačního pole 1 zadáme číselnou hodnotu, která je načtena do proměnné a předána funkci. V průběhu zpracování funkce je hodnota proměnné 10 x zvětšena. Po skončení funkce je hodnota proměnné vypsána do editačního pole 2.

Z výsledku je zřejmé, že při volání hodnotou nelze změnit původní proměnnou.

```
function vypocti(vypoctene :Integer):integer;
```



## INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Projekt: 1.5, Registrační číslo: CZ.1.07/1.5.00/34.0304

```
begin
  vypoctene := vypoctene*10;
end;
procedure TForm1.Button1Click(Sender: TObject);
var cislo:Integer;
begin
  cislo := StrToInt(Edit1.Text);
  vypocti(cislo);
  edit2.Text := IntToStr(cislo);
end;
end.
```

### Parametry volané odkazem

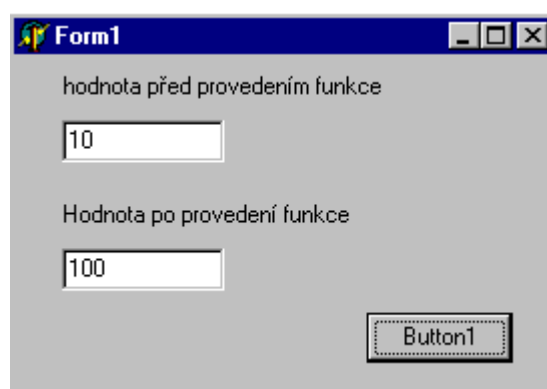
Proměnná je předána funkci tak, že zevnitř funkce je možné měnit hodnotu „externí“ proměnné. Deklarace parametru volaného odkazem musí obsahovat klíčové slovo *var*.

Na místě skutečného parametru musí být proměnná, nikoliv konstanta nebo výraz! Volání odkazem se používá pro předávání výstupních hodnot funkce nebo procedury.

### Příklad

V předchozím příkladu upravíme pouze hlavičku funkce takto:

```
function vypocti (vypoctene :Integer):integer;
```



Je zřejmé, že lze uvnitř funkce měnit hodnotu proměnné.

### Zdroje:

BINZINGER, Thomas. *Naučte se programovat v Delphi. Podrobný průvodce začínajícího uživatele*. 1. vyd. Praha: Grada, 1998, 342 s. ISBN 80-716-9685-4.

BORLAND INTERNATIONAL, Inc. *Borland Delphi 3 for Windows 95 and Windows NT: User's Guide*. Borland International, Inc., 1997.



## INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Projekt: 1.5, Registrační číslo: CZ.1.07/1.5.00/34.0304

INPRISE CORPORATION. Borland Delphi Standard 5.0 [software]. [přístup 30.12.2012].  
Dostupné z: <http://www.borland.com>. Požadavky na systém: Pentium 90 or faster (Pentium 166 recommended) Microsoft Windows 95, 98, or NT 4.0 with Service Pack 3 or later, 32 MB RAM (64 MB recommended), 55 MB for compact installation; 120 MB for full installation

PÍSEK, Slavoj. *Začínáme programovat v Delphi: podrobný průvodce začínajícího uživatele*. 1. vyd. Praha: Grada, 2000, 303 s. ISBN 80-247-9008-4.