



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

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

Reálná čísla

Představují čísla v pohyblivé řadové čárce, která se označují jako REAL. Číslo je rozděleno na mantisu a exponent. V mantise jsou uvedena vlastní desetinná místa, exponent uvádí jak daleko a kterým směrem je potřeba posunout desetinnou čárku.

Type	Rozsah	Formát v bytech
Real48	$2.9 \times 10^{-39} \dots 1.7 \times 10^{38}$	6
Single	$1.5 \times 10^{-45} \dots 3.4 \times 10^{38}$	4
Double	$5.0 \times 10^{-324} \dots 1.7 \times 10^{308}$	8
Extended	$3.6 \times 10^{-4951} \dots 1.1 \times 10^{4932}$	10
Comp	$-2^{63+1} \dots 2^{63} - 1$	8
Currency	$-922337203685477.5808 \dots 922337203685477.5807$	8

Základní REAL odpovídá typu DOUBLE.

Pro datový typ Real jsou definovány:

- aritmetické operace : +, -, *, / (dělení)
- relační operace jako u Integer

Některé důležité aritmetické rutiny:

INT	celočíslná část argumentu	INT(3.758) je 3
PI	Ludolfovo číslo	
ROUND	zaokrouhlení reálného argumentu na celé číslo	ROUND(3.785) je 4
SQR	druhá mocnina argumentu	
SQRT	druhá odmocnina argumentu	
TRUNC	odřízne desetinná místa reálného argumentu a vrací celé číslo	
SIN, COS	goniometrické funkce sinus a cosinus	
FloatToStr	převádí reálné číslo na řetězec	
StrToFloat	převádí řetězec na reálné číslo	

Příklad

Na jednoduchém příkladu si ukážeme práci s reálnými čísly. Na formulář vložíme komponenty *Label* a *Button*. Po stisknutí tlačítka se do komponenty *Label* vypíše hodnota proměnné, kterou zadáme přířazovacím příkazem např. `a:=3.4578`

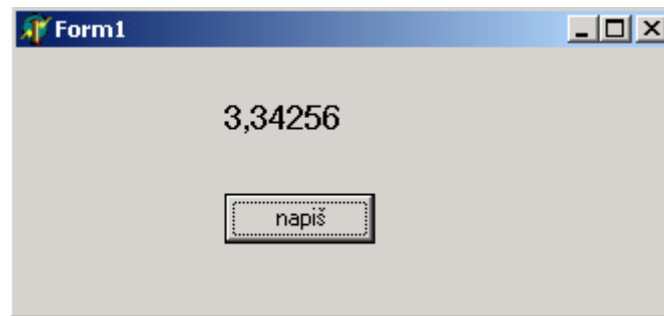


INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

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

Po stisknutí tlačítka se vypíše stejná hodnota; místo tečky se vypíše čárka.

- přiřadíte-li proměnné a hodnotu výrazu $1/3$ vypíše se 0,333333333333333
- na stejném příkladu ověřte aritmetické rutiny INT, ROUND a SQRT



Pro formátování čísel se používají ještě rutiny STR a VAL, které usnadňují zápisy čísel např. do tabulek. Procedura STR převádí číslo na řetězec.

Příklad

Upravte předchozí příklad použitím STR a ověřte různé způsoby výpisu, které jsou uvedeny v následujícím textu.

```
procedure TForm1.Button1Click(Sender: TObject);
var a: real;
    s: string;
begin
  a := (1/3);
  str(a:3:2,s);
  label1.Caption := s;
end;
end.
```

Numerický argument může obsahovat počet vystupujících znaků (šířku) a u reálných argumentů ještě počet desetinných míst (a:3:2,s).

Při výpisu celých čísel je tak umožněn zápis jednotlivých řádů pod sebe. Pokud je např. počet znaků 4 a řetězec obsahuje pouze jednu číslici, jsou před ni přidány 3 mezery.

```
i:= 5;
```

```
str(i,s)           výpis „5”
```

```
str(i:4,s)         výpis „...5”
```

U reálných čísel je výpis bez udání šířky a počtu míst uveden v exponenciálním tvaru; zkrácení výpisu se dosáhne uvedením těchto parametrů.



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

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

a:= 1/3 ;

str(a,s) výpis „3333333333333333E-0001”

str(a:3,s) výpis „0,3E-0001”

str(a:3:2,s) výpis „0,33”

Příklad

Při nákupu ve směnárně zaplatíme podle platného kurzu za 1. Euro 25,62 Kč. Do editačního pole zadáte požadovanou částku v Euro a pomocí Label vypíšete, kolik za ni zaplatíte Kč.

V příkladu použijeme konstantu, kterou bude platný kurz. (Ten vyhledáme na internetu nejlépe na webu České obchodní banky.).

Používání konstant usnadňuje editaci programu. Pokud používáme místo konstanty přímo hodnotu na různých místech programu, pak musíme při její změně všechny hodnoty v programu změnit. Při použití konstanty stačí pouze změnit její hodnotu v deklaraci.

var

Form1: TForm1;

const kurz: Real = 25.62;

implementation

*{ \$R *.DFM }*

procedure TForm1.Button1Click(Sender: TObject);

var euro, koruny : real;

begin

euro := StrToFloat(Edit1.Text);

*koruny := kurz*euro;*

Edit2.Text := FloatToStr(koruny);

end;

end.

Příklad

Vypočtete objem válce z hodnot zadaných do editačních polí.



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

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

V příkladu použijeme nastavení nápověd pro jednotlivá editační pole (*hint*) a pomocí *TabStop*.

Nastavíme, na která pole lze skákat pomocí tabelátoru a jejich pořadí.

var

Form1: TForm1;

const PI : real = 3.14;

implementation

*{ \$R *.DFM }*

procedure TForm1.Button1Click(Sender: TObject);

var polomer, vyska, objem : real;

begin

polomer := StrToFloat(Edit1.Text);

vyska := StrToFloat(Edit2.Text);

*objem := PI*SQR(polomer)*vyska;*

Edit3.Text := FloatToStr(objem);

end;

end.

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.

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