



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

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

Znaky

CHAR – Znak

Každý znak zabírá 1 bajt; je tedy možné vytvořit 256 kombinací, kterými lze definovat jednotlivé znaky. Přiřazení znaků jednotlivým číslům udává tabulka ASCII.

Přiřazení znakové proměnné můžeme provést takto:

Var a: char;

....

a:='B';

Nebo pomocí symbolu # zadání přímo ASCII kódu znaku:

c:= #66;

Přiřazení apostrofů lze provést pomocí dvojice apostrofů nebo vložením kódu:

c:= "" c:= #39

Datový typ CHAR má dvě důležité funkce:

1. ORD – vrací odpovídající pořadové číslo znaku: tj. kód
Např. **Ord('A')** je 65
2. CHR – z argumentu ve formě celého čísla vrací odpovídající znak
Např. **Chr(65)** je 'A'

Funkce Ord a Chr jsou definovány pro ordinální datové typy, jakými jsou kromě typů znak také celá čísla a výčtové typy. Jde o uspořádané množiny znaků se stanoveným pořadím, kde platí relační vztahy např.:

4 < 5

A < B

d < D (d je v ASCII tabulce před D)

U těchto uspořádaných množin jsou definovány další dvě funkce a to předchůdce PRED a následovník SUCC.

Pred(a) výsledek je předchozí hodnota proměnné a

Succ(a) výsledek je následující hodnota a



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

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

Příklady:

1. Pred('B') výsledek je A
2. Succ('A') výsledek je B
3. Ord(Succ(c)) výsledek je Ord(a) +1
4. Ord(Pred(a)) výsledek je Ord(a) -1

Příklad

Vytvoříme program pro výpis kódů stisknutých znaků. Na formulář vložíme pouze komponentu Label a využijeme událost *OnKeyPress* formuláře. Proměnné Key je v rutině předáván právě stisknutý znak. S využitím funkce Ord vypíšeme jeho pořadové číslo do Labelu.

```
procedure TForm1.FormKeyPress(Sender: TObject; var Key: Char);
begin
  Label1.Caption := IntToStr(ord(key));
end;
```

Příklad

V následujícím příkladu vyzkoušejte rutiny pro práci s ordinálními hodnotami



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

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

var

Form1: TForm1;

X,N :Integer;

implementation

*{SR *.DFM}*

procedure TForm1.BitBtn1Click(Sender: TObject);

begin

if Edit1.Text<>'' then X:=StrToInt(Edit1.Text);

if Edit9.Text<>'' then N:=StrToInt(Edit9.Text);

if N<>0 then begin Edit2.Text := IntToStr(X-N);

Edit3.Text := IntToStr(X+N);

end

else begin Edit2.Text := IntToStr(X-1);

Edit3.Text := IntToStr(X+1);

end;

if ODD(x) then edit4.Text := 'Liché' else Edit4.Text := '---';

Edit5.Text := IntToStr(Pred(X));

Edit6.Text := IntToStr(Succ(x));

Edit7.Text := Chr(x);

Edit8.Text := IntToStr(ord(x));

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

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