**Katolícka univerzita v Ružomberku**

**Pedagogická fakulta**

**Procedúry v programovacom jazyku Pascal**

**Didaktický projekt k predmetu Didaktika informatiky 2**

**Vypracoval:** Dominika Račeková **Študijný odbor:** Geografia-informatika

**Akademický rok:** 2013/14  **Ročník:** prvý (mgr.)

**Autor:** Dominika Račeková, dominika.racekova17mail.com

**Dátum odovzdania:** 28.4.2014

**Téma:** Polia

**Cieľová skupina:** 3. ročník gymnázia

**Potrebný rozsah v hodinách:** 4 vyučovacie hodiny

**Potrebné HW a SW podmienky:** počítačová učebňa s projektorom, Dev-Pascal.

**Miesto témy medzi ostatnými témami:** téma projektu nadväzuje na tému Procedúry a po nej nasleduje téma Typ záznam

**Vstupné požiadavky na vedomosti, zručnosti, návyky:** základné algoritmické myslenie, ovládanie základných príkazov jazyka Pascal

**Vyučovacie ciele – výstupné vedomosti, zručnosti a návyky:** poznať základné charakteristiky a využitie polí, vytvoriť polia, implementovať pole do programu

### Štruktúrovaný údajový typ ARRAY - POLE

Základné vlastnosti:

1. pevne daný počet prvkov,
2. všetky prvky pola sú rovnakého dátového typu,
3. jednotlivé prvky poľa majú svoje jednoznačné označenie - *index*, pričom platí, že ordinálna hodnota index-u nasledujúceho prvku je vždy o jedno väčšia ako ordinálna hodnota index-u jemu predchádzajúceho prvku.

K jednotlivým položkám poľa pristupujeme práve cez index.

Polia môžu byť jedno- a viacrozmerné.

#### Jednorozmerné polia

**Jednorozmerné pole** je rad rovnakých vlastností, ktoré sú popísané jednym identifikátorom (nazvom poľa).

Niekedy ich nazývame aj **vektory**.

Deklarácia:

type pole\_cisel=array[1..100] of real;

{ pripadne aj }

const rozsah=1..100;

type pole\_cisel=array[rozsah] of real;

{ este treba k polu zadeklarovat premennu daneho typu (pole) }

var a:pole\_cisel;

{ index typu CHAR, deklarovanie premennej typu pole v jednom prikaze }

var znaky:array['a'..'z'] of boolean;

...

znak['a']:=true; { priradnie a-temu prvku hodnotu TRUE }

#### Pole znakov

**Pole znakov** slúži na uchovávanie postupnosti znakov. Ko poľu znakov možno priradiť reťazec. Podmienkou však je, aby deklarovaný počet znakov presne zodpovedal počtu znakov v reťazci.

Pole znakov má od ostatných štruktúrovaných údajových typov výhodu v tom, že ho možno priamo vypisovať pomocou procedúr **write**,**writeln**.

Napríklad:

var a:array[1..5] of char;

begin

 a:='Janko';

 for i:=1 to 5 do read(a[i]);

 writeln(a);

...

#### Dvojrozmerné polia

Dvojrozmerné pole sa dá inak nazvať aj **tabuľka**, či **matica**. Hovoríme, že dvojrozmerné pole je aj vektor vektorov, alebo pole jednorozmerných polí.

Deklarácia je pomocou dvoch indexov.

Jednotlivé prvky poľa sú prístupné prostredníctvom hodnôt, ktoré sme použili na označenie jednotlivých rozmerov (riadkov, stĺpcov).

Príklad deklarácie:

type pocet\_radov=1..8;

 pocet\_stlpcov='A'..'H';

 sachovnica1=array[pocet\_stlpcov,pocet\_radov] of char;

 sachovnica2=array['A'..'H',1..8] of char;

 sachovnica3=array['A'..'H'] of array[1..8] of char;

var T1:sachovnica1;

 T2:sachovnica2;

 T3:sachovnica3;

**Zadania úloh**

**Úloha č. 1:** Napíš program, v ktorom počítač naplní pole náhodnými číslami. Veľkosť poľa vopred načíta z klávesnice. Nakoniec program vypíše prvky poľa na obrazovku.

**Možné riešenie:**

Program Jednorozmerne pole;

Uses CRT;

Var i,p:integer;

Cisla :array [1..20] of integer;

Begin

 Clrscr;

 Randomize;

 Writeln (‘ Zadaj pocet cisel, max. 20‘ );

 Readln (p);

 For i:=1 to p do

 Cisla [i]:= random (15); (\*vygeneruje p nahodnych cisel od 0 do 14\*)

 Writeln (‘ Vymyslene cisla su: ‘);

 For i:=1 to p do

 Writeln (i:3,‘ . cislo: ‘, cisla [i]:5);

 Repeat until keypressed;

End.

**Úloha č. 2:** Napíšte program, v ktorom zo vstupu načítate pole celých čísel a vytlačíte ho v opačnom poradí. Počet čísel na vstupe nebude viac ako 100.

**Možné riešenie:**

program opacny\_vypis;

const maxp=100;

 (\*konštanta udávajúca maximálny počet položiek poľa\*)

type vectorc=array[1..maxP] of real;

 (\*definícia vlastného typu s názvom vectorc\*)

var pocet,i:integer; (\*deklarácia premenných\*)

 cisla:vectorc;

begin

 (\* cyklom obmedzíme zadanie len maximálne 100 pre rozsah pola\*)

 repeat

 writeln('Zadaj pocet prvkov');

 readln(pocet);

 if ((pocet<1) or (pocet>100)) then

 writeln('Chybne zadany pocet cisel na nacitavanie');

 until ((pocet>=1) and (pocet<=max));

  (\* nacitanie cisel do pola daneho rozsahu\*)

 for i:=1 to pocet do (\*nacitanie prvkov pola s nazvom cisla\*)

 begin

 writeln(i,'.cislo');

 readln(cisla[i]);

 end;

 writeln('Vypis cisel v opacnom poradi');

 for i:=pocet downto 1 do (\*vypis na obr. V opacnom poradi\*)

 writeln(cisla[i]);

 readln;

end.

**Úloha č. 3:** Napíšte program, ktorý zistí, koľkokrát sa ktoré veľké písmeno anglickej abecedy vyskytne vo vstupnej postupnosti. Načítavanie je ukončené znakom hviezdička **\***.

**Možné riešenie:**

program velke\_pismena;

type velke='A'..'Z';

 (\*interval, v ktorom lezia velke pismena angl. abecedy\*)

 polepocit=array[velke] of integer;

 (\*definovany typ rozmeru A..Z pozostavajuci z celych cisel udavajucich pocetnost\*)

var pocty:polepocit;

 znak:char;

 i:integer;

begin

 for znak:='A' to 'Z' do pocty [znak]:=0;

 (\*vynuluje pocitadla pocetnosti jednotlivych pismen\*)

 writeln('Zadaj text ukonceny znakom \*');

 read(znak); (\* prvy znak \*)

 while (znak<>'\*') do

 (\*cyklus na nacitavanie znakov z klavesnice\*)

 begin

 if ((znak>='A') and (znak<='Z')) then

 pocty[znak]:=pocty[znak]+1;

 read(znak);

 end;

 writeln('Pocetnost velkych pismen je:');

 for znak:='A' to 'Z' do

begin

 if (pocty[znak]>0) then

 writeln(znak,'- ',pocty[znak]);

end;

 readln;

end.

**Spätná väzba**

**Previerková úloha:** Napíšte program, ktorý zistí, koľkokrát sa ktoré veľké písmeno anglickej abecedy vyskytne vo vstupnej postupnosti. Načítavanie je ukončené znakom hviezdička **\***.

V nasledujúcom zdrojovom kóde programu je potrebné vykonať tieto **čiastkové úlohy**:

1. V časti deklarácie typov a premenných programu je 7 chýb – chyby nájdite a opravte.
2. Zapíšte zvyšok chýbajúceho kódu

**Zdrojový kód (neúplný a čiastočne chybný) k previerkovej úlohe:**

program velke\_pismena;

type velke='A'..'Z';

 (\*interval, v ktorom lezia velke pismena angl. abecedy\*)

 polepocit=field [velke] of word;

 (\*definovany typ rozmeru A..Z pozostavajuci z celych cisel udavajucich pocetnost\*)

var pocty:polepocit;

 znak=char;

 i:array;

begin

 for znak:='K' to 'Z' do pocty [znak]:0;

 (\*vynuluje pocitadla pocetnosti jednotlivych pismen\*)

 writeln('Zadaj text ukonceny znakom \*');

 read(znak); (\* prvy znak \*)

 while (znak='\*') do

 (\*cyklus na nacitavanie znakov z klavesnice\*)

 begin

 .......

 end;

 writeln('Pocetnost velkych pismen je:');

 for znak:='A' to 'Z' do

begin

 ......

end.

**Spôsob riešenia úlohy:** písomný s možnosťou ladenia programu na počítači

**Pridelenie bodov k jednotlivým čiastkovým úlohám:**

* **1. úloha –** 7 bodov (za každú nájdenú a správne opravenú chybu 1 bod)
* **2. úloha –** 7 bodov (za každú správne doplnenú chýbajúcu časť kódu 1 bod)

**Stupnica hodnotenia:**

14 – 13 bodov:1

12 – 10 bodov: 2

9 – 7 bodov: 3

7 – 5 bodov: 4

5 – 0 bodov: 5

**Čas, vyhradený na riešenie previerkovej úlohy:** 1 vyučovacia hodina