

Typ pole

Deklarujeme:

var

```
meno pola : array [typ indexu] of typ_prvkov;
```

Pole sa skladá z veľa "jednoduchších" premenných, tzv. prvkov pola. Všetky tieto prvky sú rovnakého typu. Pristupujeme k nim pomocou, tzv. **indexu**. Typom indexu môže byť ľubovoľný ordinálny typ. Typom prvkov pola môže byť ľubovoľný typ. Štruktúra pole v pamäti zaberá toľko miesta, koľko zaberá jeden prvok krát počet prvkov pola, t.j. počet rôznych hodnôt typu indexu. Napr.

```
var
```

```
MojePole : array [1..100] of Integer;
```

definuje pole, ktoré bude mať 100 prvkov (100 celočíselných premenných)

Ku každému prvku pristupujeme pomocou indexu, t.j. pomocou hodnoty z prípustného intervalu podľa deklarácie - v našom prípade

indexom musí byť hodnota z intervalu 1..100. Ak chceme pracovať so samostatným prvkom, index zapisujeme do hranatých zátvoriek, napr.

```
P[5] := 37;
```

zmení obsah 5-teho prvku. S prvkami pola pracujeme úplne rovnako ako s obyčajnými premennými (okrem toho, že ich nesmieme použiť ako premennú for-cyklu).

Príklad 1:

Ukážeme program, ktorý bude pracovať so 7-prvkovým polom reálnych čísel. Napr. týchto 7 čísel môže znamenať 7 nameraných teplôt v priebehu jedného týždňa:

```
var
```

```
Pole: array [1..7] of Integer;
```

```
I: Integer;
```

```
begin
```

```
Pole[1] := 22;
```

```
Pole[2] := 17;
```

```
Pole[3] := 18;
```

```
Pole[4] := 19;
```

```
Pole[5] := 20;
```

```
Pole[6] := 22;
```

```
Pole[7] := 24;
```

```
for I := 1 to 7 do
```

```
    Image1.Canvas.Textout(10,14*I, Inttostr(Pole[I]));
```

```
End;
```

Vidíme, ako do prvkov pola najprv priradíme nejaké hodnoty a ako potom pomocou for-cyklu tieto hodnoty vypisujeme. Vo for-cykle sme využili počítadlo cyklu - premennú **I** na indexovanie prvkov pola.

Príklad 2:

Ak počet prvkov nie je veľký, môžeme tieto hodnoty priradiť nielen pomocou príkazu priradenia, ale môžeme priamo inicializovať obsah pola už počas deklarácie. Vtedy do okrúhlych zátvoriek vymenujeme toľko konštánt, koľko má prvkov zadeklarované pole. Napr.

```
var
```

```
Pole: array [1..7] of Integer = (22, 17, 18, 19, 20, 22, 24);
```

```
I: Integer;
```

```
Sucet, Priemer: Integer;
```

```
begin
```

```
for I := 1 to 7 do
```

```
    Image1.Canvas.Textout(10,14*I, Inttostr(Pole[I]));
```

```
Sucet := 0;
```

```
for I := 1 to 7 do
```

```
    Sucet := Sucet + Pole[I];
```

```
Priemer := Sucet Div 7;
```

```
    Image1.Canvas.Textout(10,100, 'Priemer prvkov je ' + Inttostr(Priemer));
```

```
End;
```

Zároveň sme do programu pridali aj výpočet priemernej teploty za celý týždeň.

Príklad 3:

V 100-prvkovom poli celých čísel uchováваме informácie o počte obyvateľov v 100 domoch nejakej obce. Pre jednoduchosť vygenerujeme hodnoty prvkov náhodne (ako číslo od 0 do 9) a vypíšeme tieto hodnoty:

```
var
```

```
Dom: array [1..100] of Integer;
```

```
I: Integer;
```

```
begin
```

```
for I := 1 to 100 do
```

```
    Dom[I] := Random(10);
```

```
for I := 1 to 100 do
```

```
    Image1.Canvas.Textout(10*I,10, Inttostr(Dom[I]));
```

```
End;
```

Po spustení program vypíše 100 náhodných čísel.

Príklad 4:

Nasledujúci program zistí nielen počet obyvateľov v celej obci, ale aj počet prázdnych domov (s nulovou hodnotou).

```
var
```

```
Dom: array [1..100] of Integer;
```

```
I, Pocet, Pocet0: Integer;
```

```
begin
```

```
for I := 1 to 100 do
```

```
    Dom[I] := Random(10);
```

```
for I := 1 to 100 do
```

```
    begin
```

```

    Image1.Canvas.Textout(10*I,10,Inttostr(Dom[I]));
    Pocet:=Pocet+ Dom[I]);
    if Dom[I] = 0 then Pocet0:=Pocet0+ 1;
end;
Image1.Canvas.Textout(10,100,'Pocet obyvateľov je '+Inttostr(Pocet));
Image1.Canvas.Textout (10,120,'Pocet prazdnych domov je '+Inttostr(Pocet0);
end;

```

Úlohy:

- 1) Vygenerujte náhodne N – prvkové pole zložené z celých čísel.
 - a) Vypíšte prvky poľa na obrazovku.
 - b) Vypíšte prvky na obrazovku tak, že v prvom riadku budú prvky, ktoré sú na nepárnych miestach poľa a v druhom riadku prvky na párnych miestach.
 - c) Vypíšte prvky na obrazovku tak, že párne hodnoty sú vypísané červenou farbou a nepárne zelenou.
- 2) a vstupe je neutriedené pole predstavujúce výdavky na obedy v každom týždni po celý jeden rok. Napíšte program, ktorý nájde týždeň, kedy boli výdavky najmenšie.
- 3) V apríli meriame na poludnie každý deň teplotu. Namerané hodnoty sú v rozsahu od 0 do 25. Zapišeme ich do poľa.
 - a) Nakreslite stĺpcový graf – histogram z farebných obdĺžnikov tak, že ich výška predstavuje dennú teplotu, šírku majú rovnakú. (Rovnomerne rozdeľte os x na 30 častí)
 - b) Do každého obdĺžnika vypíšte príslušnú teplotu.
 - c) Zistite priemer teplôt.
 - d) Teploty, ktoré presahujú priemer v grafe vyznačte inou farbou.
 - e) Zistite najväčšiu a najmenšiu teplotu.
 - f) V grafe zakreslite červenou najmenšiu a najväčšiu teplotu.
- 4) Na ploche obrázka simulujte sneženie. Vygenerujte y - ové pozície vločiek a uložte ich do poľa. X – ové súradnice sa budú líšiť o pravidelne o 10. Nakreslite. (vločka = znak hviezdička)
 - a) Každú sekundu všetky vločky padnú o náhodnú hodnotu smerom dolu.