



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



**OP Vzdělávání
pro konkurenceschopnost**

INVESTICE
DO ROZVOJE
VZDĚLÁVÁNÍ

Střední průmyslová škola a Vyšší odborná škola technická Brno, Sokolská 1

Šablona: Inovace a zkvalitnění výuky prostřednictvím ICT

Název: Základy programování a algoritmizace úloh

Téma: Pole

Autor: Ing. Hodál Jaroslav, Ph.D.

Číslo: VY_32_INOVACE_26-01

Anotace: *Materiál popisuje strukturovaný datový typ pole a základní principy práce s ním.
Materiál je určen pro 3. a 4. ročník oboru strojírenství a technické lyceum.
Vytvořeno: listopad 2012.*

1. Pole

- pole patří mezi **strukturované** datové typy (viz 25-06 - Datové typy v Pascalu)
- umožňuje uchovat v jedné proměnné **více informací**
- informace musí být **stejného druhu**
- **záleží na pořadí** těchto informací

- typickým příkladem informace, která má povahu pole je **vektor**
- k deklaraci proměnné typu pole se váže slůvko **array**

Deklarace polí

- pro nadeklarování proměnné typu pole už nestačí pouze název datového typu, je potřeba přidat dodatečné informace, které upřesní **typ**, **počet** a **označení** prvků pole
- deklarace může vypadat např. takto:

```
var
```

```
    a: array [1..10] of byte;
```

- za slůvkem **array** následuje v hranatých závorkách rozsah označení (**indexů**) jednotlivých prvků (zde bude mít pole 10 prvků, které budou označeny indexy od 1 do 10)
- indexy musí být vždy informace ordinálního typu (zde celé číslo)
- dále v deklaraci následuje slůvko **of** a za ním datový typ jednotlivých prvků (zde **byte**)

Práce s poli

- protože se jedná o strukturovaný datový typ, nelze na pole použít postupy, které jsou obvyklé při práci s jednoduchými údaji
- například načtení prvků do pole **nelze** provést příkazem `readln(a)`, příkaz `writeln(a)` pro výpis pole na obrazovku také fungovat nebude
- pokud tedy chceme pracovat s polem, musíme pracovat s jeho **jednotlivými prvky**, jako by to byly samostatné jednoduché proměnné
- tuto práci nám umožní následující zápis:

```
readln(a[1]) ; writeln(a[5]) ; a[3] := 0 ;
```

Práce s poli – typické konstrukce (1)

- index v zápise `a[1]` může být nahrazen i proměnnou (což například u názvu proměnné `b1` nejde)
- možnost nahradit index proměnnou otevírá cestu k použití **cyklů** řízených proměnnou (viz 25-15) pro práci s prvky pole
- typickou konstrukcí je např. načtení všech prvků do pole

```
var a:array[1..10] of byte;
...
for i := 1 to 10 do
begin
    write('Zadej ', i, ' prvek: ')
    readln(a[i]);
end;
```

Práce s poli – typické konstrukce (2)

- další typickou konstrukcí je výpis pole

```
var a:array[1..10] of byte;  
...  
for i := 1 to 10 do  
  writeln(a[i]);
```

- nebo také nastavení prvků pole na zvolenou hodnotu (obvykle 0 např. při práci s polem počítadel)

```
var a:array[1..10] of byte;  
...  
for i := 1 to 10 do  
  a[i] := 0;
```