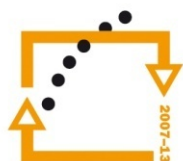




MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



**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: Množina

Autor: Ing. Hodál Jaroslav, Ph.D.

Číslo: VY_32_INOVACE_26–11

Anotace: *Materiál popisuje definici a použití datového typu množina v jazyce Pascal.
Materiál je určen pro 3. a 4. ročník oboru strojírenství a technické lyceum.
Vytvořeno: březen 2013.*

11. Množina

- datový typ množina patří mezi strukturované datové typy
- v jedné proměnné tedy můžeme mít **více informací**
- **nezáleží na pořadí** informací v proměnné
- informace obsažené v množině musí být všechny **stejného typu** (tomuto typu se pak říká **bázový typ**)
- bázový typ množiny musí být ordinální datový typ, který používá pořadová čísla z rozsahu 0-255 (např. `char`, `byte`; nemůže být ale např. `word` nebo dokonce `real`)
- často je bázový typ množiny **výčtový datový typ** (viz 25-06)
- množiny také patří mezi uživatelsky definovatelné typy – programátor si je může vytvořit dle svých potřeb

Použití množin v programu

- proměnnou typu množina můžeme použít pro uchovávání informací o výskytu několika nezávislých jevů (lze je chápat jako stavové přepínače), přičemž každá informace zabírá pouze jediný bit
- existují dva typické způsoby jak pracovat s množinami
- buď potřebujeme **proměnnou** typu množina, ve které uchováváme informace nebo potřebujeme jednorázově použít ve zdrojovém kódu **konstantní množinu**
- pomocí množiny se například reprezentuje použití či nepoužití písmen pro diskové jednotky v počítači
- tato množina zabírá v paměti 32 bitů a každý z prvních 26 bitů svojí hodnotou 0/1 odpovídá přítomnosti jednotky A-Z

Deklarace proměnné typu množina

```
type
```

```
    FunkcniKlavesy=(ctrl, alt, shift);
```

```
var
```

```
    Stisknuto : set of FunkcniKlavesy;
```

- uvedený zápis obsahuje definici výčtového typu `FuncniKlavesy` a deklaraci proměnné `Stisknuto`, která může obsahovat množinu, jejímiž prvky mohou být libovolné kombinace hodnot typu `FuncniKlavesy`
- nese-li proměnná `Stisknuto` informaci o stavu stisknutí funkčních kláves při nějaké události, je vhodné ji reprezentovat jako množinu, protože může být stisknuta kterákoliv z nich, několik zároveň nebo také žádná

Přímý zápis množiny ve zdrojovém kódu

- pokud bychom potřebovali přímo jednorázově zapsat množinu do zdrojového kódu programu, použijeme tzv. konstruktor množiny, který je považován za výraz

```
if c in ['A'..'Z'] then  
    writeln('velké písmeno');
```

```
if c in ['a','e','i','o','u','y'] then  
    writeln('malá samohláska');
```

```
if Stisknuto = [] then writeln('nebyla  
stisknuta funkční klávesa');
```