

# PROGRAMOVÁNÍ

Ing. Petr Stránský

# PASCAL

## DATOVÝ TYP INTEGER

VY\_32\_INOVACE\_08\_1\_08\_PR



evropský  
sociální  
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,  
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání  
pro konkurenceschopnost

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

# Definice

Typ *integer* specifikuje **konečnou souvislou podmnožinu celých čísel**.

Maximální absolutní hodnota čísel typu *integer* je definována standardním identifikátorem konstanty **MAXINT** typu *integer*. Pro libovolnou hodnotu *n* typu *integer* tedy platí:

$$- \text{maxint} \leq n \leq \text{maxint}$$

Např.: v Delphi verze 6 je hodnota *maxint* rovna  $2\,147\,483\,647 = 2^{31}-1$

# Definované operace I

| operace    | význam                        | typ výsledku |
|------------|-------------------------------|--------------|
| +          | sčítání                       | integer      |
| -          | odčítání nebo změna znaménka  | integer      |
| *          | násobení                      | integer      |
| <b>div</b> | celočíslné dělení             | integer      |
| <b>mod</b> | zbytek po celočíselném dělení | integer      |
| <b>abs</b> | absolutní hodnota čísla       | integer      |
| <b>sqr</b> | druhá mocnina                 | integer      |
| <b>odd</b> | lichost čísla                 | boolean      |

Je nutné zabezpečit, aby výsledek libovolné operace nepřesáhl číslo *maxint*. V případě přesáhnutí čísla *maxint* se výsledek překlopí do záporných čísel.

# Definované operace II

| operace | význam           | typ výsledku |
|---------|------------------|--------------|
| =       | rovnost čísel    | boolean      |
| <>      | nerovnost čísel  | boolean      |
| <       | menší než        | boolean      |
| >       | větší než        | boolean      |
| >=      | větší nebo rovno | boolean      |
| <=      | menší nebo rovno | boolean      |

Je nutné zabezpečit, aby výsledek libovolné operace nepřesáhl číslo *maxint*.  
V případě přesáhnutí čísla *maxint* se výsledek překlopí do záporných čísel.

# Příklady I

| příkaz          | výsledek                                                  |
|-----------------|-----------------------------------------------------------|
| <b>10 + 20</b>  | 30                                                        |
| <b>20 - 15</b>  | 5                                                         |
| <b>5 * 6</b>    | 30                                                        |
| <b>18 div 6</b> | 3                                                         |
| <b>20 div 6</b> | 3 !!!!! celočíselné dělení, výsledkem je celé číslo !!!!! |
| <b>15 mod 5</b> | 0 (zbytek po celočíselném dělení)                         |
| <b>16 mod 5</b> | 1 (zbytek po celočíselném dělení)                         |

# Příklady II

| příkaz              | výsledek |
|---------------------|----------|
| <b>5 = 5</b>        | TRUE     |
| <b>6 &lt;&gt; 4</b> | TRUE     |
| <b>7 &lt; 5</b>     | FALSE    |
| <b>10 &gt; 5</b>    | TRUE     |
| <b>15 &gt;= 10</b>  | TRUE     |
| <b>15 &lt;= 15</b>  | TRUE     |

# Příklady III

| příkaz         | výsledek                                                              |
|----------------|-----------------------------------------------------------------------|
| <b>abs(-5)</b> | 5                                                                     |
| <b>abs(5)</b>  | 5                                                                     |
| <b>sqr(4)</b>  | 16                                                                    |
| <b>sqr(-4)</b> | 16                                                                    |
| <b>odd(3)</b>  | TRUE (číslo je liché)                                                 |
| <b>odd(4)</b>  | FALSE (číslo není liché, je sudé)                                     |
| <b>odd(0)</b>  | FALSE !!!! (nula se v programovacích jazycích počítá mezi sudá čísla) |