

13 Relačné a logické operátory

13.1 Relačné operátory a výrazy

Relačné (**alebo tiež porovnávacie**) **operátory** sa využívajú pri vytváraní rozhodovacích podmienok v podmienených príkazoch a v cykloch. Jazyk Java poskytuje nasledovné relačné operátory:

| | |
|--------------------|--------------------|
| <code>></code> | je väčší než |
| <code>>=</code> | je väčší rovný než |
| <code><</code> | je menší než |
| <code><=</code> | je menší rovný než |
| <code>==</code> | je rovný |
| <code>!=</code> | nie je rovný |

Pomocou hore uvedených operátorov vytvárame relačné (porovnávacie) výrazy. Vyhodnotením relačného výrazu sa získa logická pravda - **true** alebo logická nepravda **false**. Operandami relačných výrazov sú aritmetické výrazy, pričom aritmetické výrazy majú vyššiu prioritu.

Príklady relačných výrazov:

```
a + 3 > b - 10           //to isté ako (a + 3) > (b - 10)
cislo1 == cislo2         //pozor na zápis cislo1 = cislo2
x >= y + Math.sqrt(z+y)
znak <= 'Z'
retazec.length() != 100
```

13.2 Logické operátory a výrazy

Logické **operátory** (rovnako ako relačné operátory) sa využívajú pri vytváraní rozhodovacích podmienok v podmienených príkazoch a v cykloch. Umožňujú vytvárať zložitejšie rozhodovacie podmienky. Jazyk Java poskytuje nasledovné logické operátory:

| | |
|-------------------------|---|
| <code>&&</code> | logický súčin so skráteným vyhodnocovaním (and) |
| <code> </code> | logický súčet so skráteným vyhodnocovaním (or) |
| <code>!</code> | negácia (not) |
| <code>&</code> | logický súčin s úplným vyhodnocovaním (and) |
| <code> </code> | logický súčet s úplným vyhodnocovaním (or) |

Vyhodnotením logického výrazu sa získa logická pravda - **true** alebo logická nepravda **false**. Operandami logických výrazov sú relačné výrazy, pričom relačné výrazy majú vyššiu prioritu. Výnimku tvorí operátor **!**, ktorý má vyššiu prioritu než všetky relačné a aritmetické operátory.

Operátory && a &

Logický výraz s operátorom **&&** vráti hodnotu **true** iba vtedy, ak sú **oba** relačné výrazy **pravdivé**. Inak vráti hodnotu **false**. Predpokladajme, že v príkladoch logických výrazov majú premenné priradené nasledovné hodnoty: a=10; b=15; c=25. Potom:

| | |
|---------------------|---|
| b >= a && b <= c | //true, lebo oba relačné výrazy sú pravdivé |
| a >= c && b <= c | //false, lebo jeden z relačných výrazov je nepravdivý |
| !(a >= c) && b <= c | //true |

V prípade použitia operátora **&&** sa logický súčin vyhodnocuje skráteno, t.j. ak prvý z operandov má hodnotu **false**, tak už je celý výraz vyhodnotený ako **false** (druhý z operandov sa nevyhodnocuje). Ak chceme aby bol logický výraz vyhodnotený úplne, tak použijeme operátor **&**.

Operátory || a |

Logický výraz s operátorom **||** vráti hodnotu **true**, ak **aspoň jeden** z relačných výrazov je **pravdivý**. Predpokladajme, že v príkladoch logických výrazov majú premenné priradené nasledovné hodnoty: a=10; b=15; c=25. Potom:

| | |
|---------------------|--|
| a > b c > b | //true, lebo c > b |
| a + b >= c b==c | //true, lebo a+b>=c |
| b < a c < a | //false, lebo oba relačné výrazy sú nepravdivé |
| c >= b - a b > a | //true, lebo oba relačné výrazy sú pravdivé |

V prípade použitia operátora **||** sa logický súčin vyhodnocuje skráteno, t.j. ak prvý z operandov má hodnotu **true**, tak už je celý výraz vyhodnotený ako **true** (druhý z operandov sa nevyhodnocuje). Ak chceme aby bol logický výraz vyhodnotený úplne, tak použijeme operátor **|**.

Operátor !

Logický operátor **!** neguje pravdivostnú hodnotu výrazu. Príklady:

```
! ( a==b)           //true, lebo a== b je false
```

```
!(a<c) //false, lebo a<c je true
```

Okrem spomínaných relačných a logických operácií ponúka Java aj bitové operácie – bitový súčin, bitový súčet, bitový exkluzívny súčet, bitový posun atď. Tieto operácie nebudeme preberať.

13.3 Otázky

1. Uved'te, ktoré relačné operátory pozná Java.
2. Aká hodnota sa získa vyhodnotením relačného výrazu?
3. Uved'te príklady relačných výrazov.
4. Uved'te, ktoré logické operátory pozná Java.
5. Uved'te príklady výrazov s logickými operátormi.
6. Uved'te v čom spočíva rozdiel medzi úplným a skráteným vyhodnocovaním logických výrazov.