EECS 1022 P Jan 18 Lecture

Yves Lesperance

Do This Week

- Quiz #1 on Wed at 10:30, see announcement on Moodle site.
- Lab Audio Generation: prepare, do pre-lab quiz, go to your lab section, and do report
- Read and digest lecture material

Relational Operators

<  <=  >  >=

==  !=

Numeric operands

Operands of any type

All relational operands are "odd" in that they violate closure: no matter what the operand type is, the result type is always boolean.
### Operator Precedence

<table>
<thead>
<tr>
<th>Precedence</th>
<th>Operator</th>
<th>Operands</th>
<th>Syntax</th>
<th>true if</th>
</tr>
</thead>
<tbody>
<tr>
<td>-7</td>
<td>&lt;</td>
<td>numeric</td>
<td>x &lt; y</td>
<td>x is less than y</td>
</tr>
<tr>
<td>-6</td>
<td>&lt;=</td>
<td>numeric</td>
<td>x &lt;= y</td>
<td>x is less than or equal to y</td>
</tr>
<tr>
<td>-5</td>
<td>&gt;</td>
<td>numeric</td>
<td>x &gt; y</td>
<td>x is greater than y</td>
</tr>
<tr>
<td></td>
<td>&gt;=</td>
<td>numeric</td>
<td>x &gt;= y</td>
<td>x is greater than or equal to y</td>
</tr>
<tr>
<td>-4</td>
<td>instanceof</td>
<td>x instanceof C</td>
<td>x is true if object reference x points at an instance of class C or a subclass of C.</td>
<td></td>
</tr>
<tr>
<td>-3</td>
<td>==</td>
<td>any type</td>
<td>x == y</td>
<td>x is equal to y</td>
</tr>
<tr>
<td>-2</td>
<td>!=</td>
<td>any type</td>
<td>x != y</td>
<td>x is not equal to y</td>
</tr>
</tbody>
</table>

### Boolean Expressions

- **Relational Expression**
  
  `(k < 0)`

- **Boolean Variable**

  ```java
  boolean b = k < 0
  ```

- **Boolean Expression**

  `(k < 0 || b && m > h)`

  Uses boolean operators: `&&`, `||`, and `!`

### Operator Precedence

- `.`
- `++` `--` `!`
- `cast`
- `*` `/` `%`
- `+` `-`
- `<` `<=` `>` `>=`
- `==` `!=`
- `&` `&`
- `|` `||`
- `=` `op=`
Examples

• Express the condition \( x \in [a, b) \)

• Express the condition \( x \notin [a, b) \)

(\( x \geq a \land x < b \))

Can use deMorgan's Law to convert negated conjunctions to disjunctions.

(\( x < a \lor x \geq b \))

Examples

• Express the condition \( x \in [a, b) \)

(\( x \geq a \land x < b \))

• Express the condition \( x \notin [a, b) \)

(! (\( x \geq a \land x < b \)))

Operations on char data

• Can use integer operators with char data, e.g.
  char let1 = 'D';
  char let2 = (char) (let1 + 1);
• Here let1 is promoted to int when 1 is added and must be cast back to char.
• What is the value of
  (let1 - 'A' + 1)?
• char has no operators so both arguments of - are promoted to int.