Logical (Boolean) Operators

Instructor: Chi Tse (Ricky)
## Logical Operators

<table>
<thead>
<tr>
<th>SYMBOL</th>
<th>OPERATION</th>
<th>C++ SYNTAX</th>
<th>ACTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;&amp;</td>
<td>AND</td>
<td><code>expr1 &amp;&amp; expr2</code></td>
<td>Evaluate <code>expr1</code>. If it’s true, evaluate <code>expr2</code>. Then, if both are true, return true; otherwise, return false.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>OR</td>
</tr>
<tr>
<td>!</td>
<td>NOT</td>
<td><code>! expr1</code></td>
<td>Evaluate <code>expr1</code>. Reverse the true/false value.</td>
</tr>
</tbody>
</table>

(from chapter 2 of textbook)
The “and” Operator &&

- We can combine two comparisons using the “and” operator.
- The symbol for “and” operator is &&.

- Syntax for a Boolean Expression using &&
  
  \((\text{comparison}_1) \text{ && } (\text{comparison}_2)\)

- The Boolean expression is true only if both comparisons are true.
The “and” Operator \\

Example: With an *if* – *else* Statement

```cpp
If ( (score > 0) && (score < 10) )
    cout << “The score is between 0 and 10 \n”;
else
    cout << “The score is not between 0 and 10 \n”;`
The “or” Operator ||

- We can also combine two comparisons using the “or” operator.
- The symbol for “or” operator is ||
- Syntax for a Boolean Expression using ||

```
(comparison_1) || (comparison_2)
```

- The Boolean expression is true if one or both of the comparisons are true.
Example: With an *if – else* Statement

```cpp
If ( (x == 1) || (x == y) )
    cout << “x is 1 or x equals to y.\n”; 
else
    cout << “x is neither 1 nor equal to y. \n”;
```