Ann has ID, or Bob has ID and Ann does not.

If it is not true that both Ann and Bob don't have ID.

To buy beer, someone must have ID.

Review Problem 3
Truth Table:

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
<th>Z</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Boolean Function: \( F = X + YZ \)

Circuit Diagram:

Representations of Boolean Functions

Readings: 2.3-2.5, 2.9-2.9.2, 4.1-4.2.7, 4.3, 4.9
number of gates (or gate packages) influences manufacturing costs.

Fewer levels of gates implies reduced signal propagation delays.

Fewer inputs implies faster gates in some technologies.

- Reduce number of levels of gates.
- Reduce number of gates.
- Reduce number of literals (gate inputs).

Logic Minimization: Reduce complexity of the gate level implementation.

\[
\overline{ABC} + \overline{AB}C + \overline{AC} + \overline{ABC} + \overline{AC} + \overline{AB}C
\]

Why Boolean Algebra/Logic Minimization?
Basic Boolean Identities:

\[ 0 = X \cdot X \]
\[ 1 = X + X \]

\[ X = X \cdot X \]
\[ X = X + X \]

\[ 0 = 0 \cdot X \]
\[ 1 = 1 + X \]

\[ X = I \cdot X \]
\[ X = 0 + X \]
(Z+X)(Y+X) = ZY+X

Z(YX) = (ZY)X

Z + (Y + X) = (Z + Y) + X

X + Y = Y + X

Basic Laws

Distributive Law:

Associative Law:

Commutative Law:
Truth Table:

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
<th>Z</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Boolean Function: \( F = \overline{XYZ} + \overline{XY} + XYZ \)

Reduce Function:

\[
= y(xz + \overline{z})
= (x + \overline{z})(z + \overline{z})
= (x + \overline{z})\overline{z}
= x \overline{z}
\]

Properties:

- Identity
- Commutative
- Distributive
\[\lambda^x = (\lambda + \underline{x})x \quad \checkmark\]
\[x = (\lambda + \underline{x})(\lambda + x) \quad \checkmark\]
\[\underline{x} = (\lambda + x)x \quad \checkmark\]
\[\lambda + \underline{x} = \lambda x + x \quad \checkmark\]
\[\underline{x} = \lambda x + \lambda x \quad \checkmark\]
\[x = \lambda x + x \quad \checkmark\]

Advanced Laws (Absorption)
Reduce Function:

\[ F = XZ + YZ \]

Boolean Function: \[ F = XYZ + XZ \]

Truth Table:

<table>
<thead>
<tr>
<th>F</th>
<th>Z</th>
<th>Y</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Boolean Manipulations (cont.)