Ex.

10,0) — (400,0)

Red

Blue

(250,100)

(10,400)

(300,50)

Processing code:
```
size(400, 400);
background(255, 0, 0);
rectMode(CENTER);
fill(0, 0, 255);
rect(175, 75, 250, 50);
```
Ex: how many pixels are in a 400x400 image?

#pixels = 400×400 = 160,000

#bytes = 160,000 ÷ 8 = 20,000

How many bytes in a 1000x1000 image?

#pixels = 1000×1000 = 1,000,000

#bytes = 1,000,000 ÷ 8 = 125,000
Logic gates

**And**

\[ \begin{array}{ccc}
 a & b & a \cdot b \\
 0 & 0 & 0 \\
 0 & 1 & 0 \\
 1 & 0 & 0 \\
 1 & 1 & 1 \\
\end{array} \]

**Or**

\[ \begin{array}{ccc}
 a & b & a + b \\
 0 & 0 & 0 \\
 0 & 1 & 1 \\
 1 & 0 & 1 \\
 1 & 1 & 1 \\
\end{array} \]

**Not**

\[ \begin{array}{ccc}
 a & \overline{a} \\
 0 & 1 \\
 1 & 0 \\
\end{array} \]
or gate from transistors:
not gate
## Exclusive Or

<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
<th>a ⊕ b</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

## Truth Table

<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
<th>( \overline{a} )</th>
<th>( \overline{b} )</th>
<th>a ( \cdot ) ( \overline{b} )</th>
<th>a ( \cdot ) b</th>
<th>(a ( \cdot ) b) + (a ( \cdot ) ( \overline{b} ))</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

\[ a \oplus b = \overline{a} \cdot b + a \cdot \overline{b} \]
Define a combinational circuit.

A combinational circuit is a collection of logic gates that transforms a set of binary inputs to a set of binary outputs.
A circuit with inputs $I_n$, $I_{n-1}$, ..., $I_1$ and outputs $O_{m-1}$, $O_{m-2}$, ..., $O_1$. The outputs are connected to logical expressions, which can be converted to Boolean expressions:

- Many to one (inputs) connecting to Circuit
- 1 to 1 (Circuit) connecting to logical expressions
- Many to one (logical expressions) connecting to truth tables

The circuits are related to logical expressions and truth tables through a 1 to 1 mapping.
Ex 1-bit compare for equality

<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
<th>out</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>