Recall that a computer is just a collection of circuits that interact to perform various tasks
- memory, input, output, processing, ....

Circuits are composed of logic gates
- gate: device that performs an operation on one or more electrical signals

Logic gates
- NOT
- AND
- OR
- XOR
- NAND
- NOR

XOR gate
- Output is 1 if either input is 1 (but not both)
- Expression: $A \oplus B$
- Truth table and logic diagram:
NAND gate

- Output is always 1 unless both inputs are 1
- Expression: \((A \cdot B)'\)
- Truth table and logic diagram:

```
<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</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>
```

NOR gate

- Output is 1 only if all inputs are 0
- Expression: \((A + B)'\)
- Truth table and logic diagram:

```
<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>X</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>0</td>
</tr>
</tbody>
</table>
```

More than 2 inputs

- Same rules apply for logic gates with more than 2 inputs
- In general: the number of rows in the truth table = \(2^n\), where \(n\) is the number of inputs

Example

```
<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</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>0</td>
</tr>
</tbody>
</table>
```
### Half-adder circuit
- Circuit used to express the results of adding two single bits to each other
- Sum = $A \oplus B$
- Carry = $AB$

### Full-adder
- Expresses the sum of two bits plus a carry bit
- Sum = $(A \oplus B) \oplus \text{Carry-in}$
- Carry-out = $AB + (A \oplus B) \text{Carry-in}$

### Mathematical laws of boolean arithmetic: AND
- Commutative: $AB = BA$
- Associative: $ABC = (AB)C = A(BC)$
- Distributive: $A(B+C) = AB + AC$
- Identity: $A1 = A$
- Complement: $AA' = 0$

### Mathematical laws of boolean arithmetic: OR
- Commutative: $A+B = B+A$
- Associative: $A+B+C = (A+B) + C = A + (B+C)$
- Distributive: $A + BC = (A+B)(A+C)$
- Identity: $A+0 = A$
- Complement: $A+A' = 1$