Propositional Logic

January 11, 2019

---

1 Propositions

- A ***Proposition*** is an atomic statement that either has the truth value $true$ or $false$
- For example:
  - $1 + 1 = 2$
  - $1 + 2 = 4$
  - The Earth is closer to the Sun than Mars
  - My eyes are green

---

2 Boolean Operators

- Negation
  - ***Negation*** is a unary boolean operator
  - This means it only takes one arugment
  - The truth table for negation is:

x	$\neg x$
$true$	$false$
$false$	$true$

- Conjunction
  - ***Conjunction*** is a binary boolean operator
  - This means it takes two arguments
  - The truth table for conjunction is:

x	y	$x \wedge y$
$false$	$false$	$false$
$false$	$true$	$false$
$true$	$false$	$false$
$true$	$true$	$true$

- Disjunction
  - ***Disjunction*** is a binary boolean operator
  - This is the truth table for disjunction:

x	y	$x \vee y$
$false$	$false$	$false$
$false$	$true$	$true$
$true$	$false$	$true$
$true$	$true$	$true$

- Exclusive-Or
  - ***Exclusive-or*** is a binary boolean operator
  - This is the truth table for exclusive-or

x	y	$x \oplus y$
$false$	$false$	$false$
$false$	$true$	$true$
$true$	$false$	$true$
$true$	$true$	$false$

- Implication
  - ***Impication*** is a binary boolean operator
  - $x \Rightarrow y$ means that when x when is $true$, y is $true$, but y can be $true$ when x is false
  - This is the truth table for implication

x	y	$x \Rightarrow y$
$false$	$false$	$true$
$false$	$true$	$true$
$true$	$false$	$false$
$true$	$true$	$true$

- Equivalence
  - ***Equivalence*** is a binary boolean operator
  - This is the truth table for equivalence

x	y	$x \Leftrightarrow y$
$false$	$false$	$true$
$false$	$true$	$false$
$true$	$false$	$false$
$true$	$true$	$true$

- Nand
  - ***Nand*** is a binary boolean operator
  - It is the negation of conjunction
  - This is the truth table for nand

x	y	$x \Uparrow y$
$false$	$false$	$true$
$false$	$true$	$true$
$true$	$false$	$true$
$true$	$true$	$false$

- Nor
  - ***Nor*** is a binary boolean operator
  - It is the negation of disjunction
  - This is the truth table for nor

x	y	$x \Downarrow y$
$false$	$false$	$true$
$false$	$true$	$true$
$true$	$false$	$true$
$true$	$true$	$false$

---

3 Boolean Formulae

- if P is the set of atomic propositions, a boolean formula bf can be derived from the following rules:
  - $bf ::= p$ *where* $p \in P$
  - $bf ::= \neg bf$
  - $bf ::= bf \ op \ bf$
  - $op ::= \wedge | \vee | \oplus | \Rightarrow | \Leftarrow | \Leftrightarrow | \Uparrow | \Downarrow$
- We will denote the set of boolean formulae defined by the above rule as F

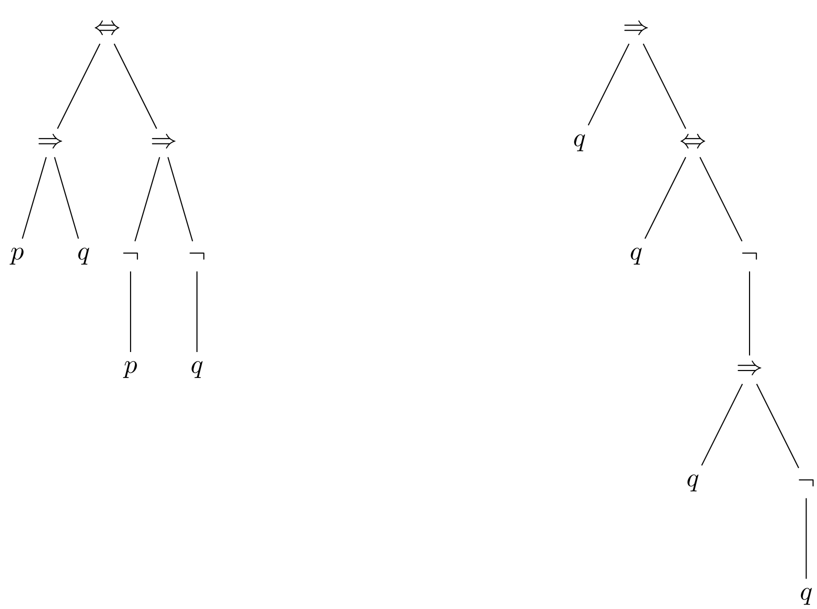
---

4 Ambiguity, Precendence and Associativity

- The previous rules by themselves are not sufficient to assign a single meaning to the following formula

$$p \Rightarrow q \Leftrightarrow \neg p \Leftrightarrow \neg q$$

We can represent this formula by at least two different trees



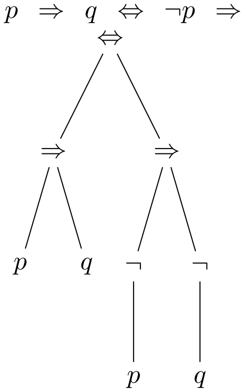
- When the same formula can have more than one meaning, the formula is *ambiguous*

- Ambiguity can be addressed by:
  - Using parentheses
  - Assigning *precedence* to the operators
  - assuming that operators *associate* from left to right
- The order of *precedence* from highest to lowest it:

$$\begin{array}{c} \neg \\ \vee,\Uparrow \\ \wedge,\Downarrow \\ \Rightarrow \\ \Leftrightarrow,\oplus \end{array}$$

So, now we have only one meaning for each of the following formulae

$p \;\Rightarrow\; q \;\Leftrightarrow\; \neg p \;\Rightarrow\; \neg q$



$p \;\Rightarrow\; (q \;\Leftrightarrow\; \neg(p \Rightarrow \neg q))$