Logica (I&E)

najaar 2017

`http://liacs.leidenuniv.nl/~vlietrvan1/logica/`

Rudy van Vliet
kamer 140 Snellius, tel. 071-527 2876
rvvliet(at)liacs(dot)nl

college 2, maandag 11 september 2017

1.4 Semantics of propositional logic
1.2 Natural deduction

Als je op balbezit speelt, hoef je niet te verdedigen want er is maar één bal.

1.4. Semantics of propositional logic

Definition 1.28.
1. The set of truth values contains two elements T and F, where T represents 'true' and F represents 'false'.
2. A *valuation* of *model* of a formula ϕ is an assignment of each propositional atom in ϕ to a truth value.

1.4. Semantics of propositional logic

Definition 1.28.
1. The set of truth values contains two elements T and F, where T represents 'true' and F represents 'false'.
2. A *valuation* of *model* of a formula ϕ is an assignment of each propositional atom in ϕ to a truth value.

Example 1.29. $\phi = p \vee \neg q$
p: F
q: T

Truth table for conjunction

ϕ	ψ	$\phi \wedge \psi$
$\ldots$	$\ldots$	$\ldots$

Truth table for conjunction

ϕ	ψ	$\phi \wedge \psi$
T	T	T
T	F	F
F	T	F
F	F	F

Truth tables

ϕ	ψ	$\phi \wedge \psi$
T	T	T
T	F	F
F	T	F
F	F	F

ϕ	ψ	$\phi \vee \psi$
T	T	T
T	F	T
F	T	T
F	F	F

ϕ	ψ	$\phi \rightarrow \psi$
...	...	...

ϕ	$\neg\phi$
T	F
F	T

Truth table for implication

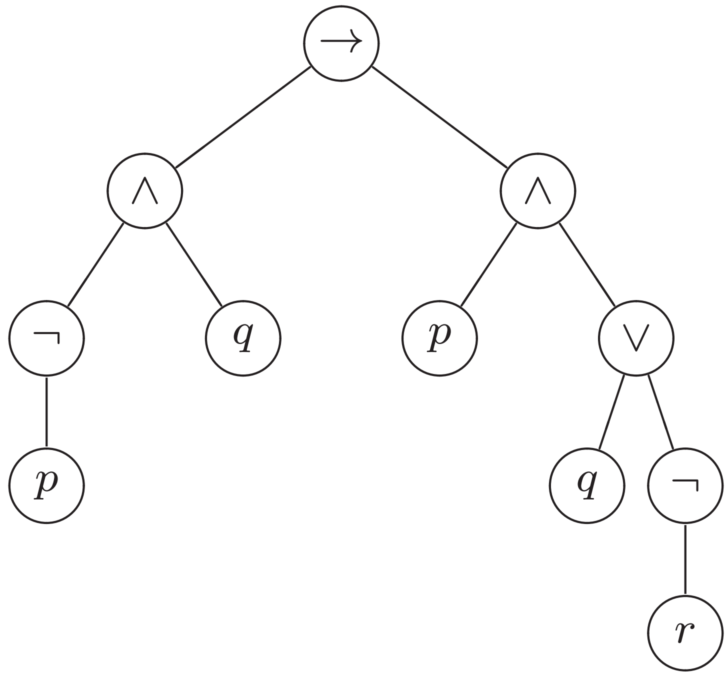
ϕ	ψ	$\phi \to \psi$
T	T	T
T	F	F
F	T	T
F	F	T

ϕ	ψ	$\neg\phi \lor \psi$
T	T	T
T	F	F
F	T	T
F	F	T

Semantically equivalent

Determining truth value in tree

$\neg p \wedge q \rightarrow p \wedge (q \vee \neg r)$



$n = 3$, so 2^3 lines in truth table

p : T $\quad$ q : F $\quad$ r : T

Determining truth value in table

$(p \rightarrow \neg q) \rightarrow (q \vee \neg p)$

p	q	$\neg p$	$\neg q$	$p \rightarrow \neg q$	$q \vee \neg p$	$(p \rightarrow \neg q) \rightarrow (q \vee \neg p)$
T	T	...	...	...	...	...
T	F	...	...	...	...	...
F	T	...	...	...	...	...
F	F	...	...	...	...	...

Determining truth value in table

$(p \to \neg q) \to (q \vee \neg p)$

p	q	$\neg p$	$\neg q$	$p \to \neg q$	$q \vee \neg p$	$(p \to \neg q) \to (q \vee \neg p)$
T	T	F	F	F	T	T
T	F	F	T	T	F	F
F	T	T	F	T	T	T
F	F	T	T	T	T	T

1.4.2. Mathematical induction

$$1 + 2 + 3 + 4 + \cdots + n = \ldots$$

Mathematical induction

For property M of natural numbers:

1. Base case: The natural number 1 has property M, i.e., we have a proof of $M(1)$

2. Inductive step: If n is a natural number which *we assume* to have property $M(n)$, then *we can show* that $n+1$ has property $M(n+1)$; i.e., we have a proof of $M(n) \rightarrow M(n+1)$.

Mathematical induction

For property M of natural numbers:

1. Base case: The natural number 1 has property M, i.e., we have a proof of $M(1)$

2. Inductive step: If n is a natural number which *we assume* to have property $M(n)$, then *we can show* that $n+1$ has property $M(n+1)$; i.e., we have a proof of $M(n) \rightarrow M(n+1)$.

Definition 1.30. The principle of mathematical induction says that, on the grounds of these two pieces of information above, every natural number n has property $M(n)$.
The assumption of $M(n)$ in the inductive step is called the *induction hypothesis*.

Natural numbers

Mathematics: $\mathbb{N} = \{1, 2, 3, 4, \ldots\}$

Computer science: $\mathbb{N} = \{0, 1, 2, 3, 4, \ldots\}$

Theorem 1.31. The sum $1+2+3+4+\cdots+n$ equals $n\cdot(n+1)/2$ for all natural numbers n.

Proof: $\text{LHS}_n = \text{RHS}_n \ldots$

Definition. Let the level of the root in a binary tree be 1, the level of the children of the root be 2, ... (N.B.: different from Algoritmiek). The *height* of a binary tree is the maximum level of the tree. A binary tree of height h is called *filled*, if every level of the tree contains the maximum number of nodes.

Exercise. Prove by induction that
(a) for each level l of a filled binary tree, the number of nodes at level l equals 2^{l-1},
(b) the number of nodes in a filled binary tree of height h equals $2^h - 1$,
(c) the maximum number of swaps needed for (bottom-up) heapify in a filled binary tree of height h equals $2^h - 1 - h$.

Variants of induction

Mathematical induction:

1. Base case: The natural number 1 has property M, i.e., we have a proof of $M(1)$

2. Inductive step: If n is a natural number which *we assume* to have property $M(n)$, then *we can show* that $n+1$ has property $M(n+1)$; i.e., we have a proof of $M(n) \to M(n+1)$.

Variants of induction

Mathematical induction:

1. Base case: The natural number 1 has property M, i.e., we have a proof of $M(1)$

2. Inductive step: If n is a natural number which *we assume* to have property $M(n)$, then *we can show* that $n+1$ has property $M(n+1)$; i.e., we have a proof of $M(n) \rightarrow M(n+1)$.

Course-of-values induction:

2. Inductive step: If n is a nonnegative, integer number for which *we assume* that $M(1) \wedge M(2) \wedge \cdots \wedge M(n)$ holds, then *we can show* that $n+1$ has property $M(n+1)$; i.e., we have a proof of $M(1) \wedge M(2) \wedge \cdots \wedge M(n) \rightarrow M(n+1)$.

Fibonacci

(variant of Exercise 1.4.8)

$F_1 = 1$,
$F_2 = 1$,
$F_{n+1} = F_n + F_{n-1}$ if $n \geq 2$

Use course-of-values induction to prove that F_n is even, if and only if $n \equiv 0 \pmod 3$.

Variants of induction

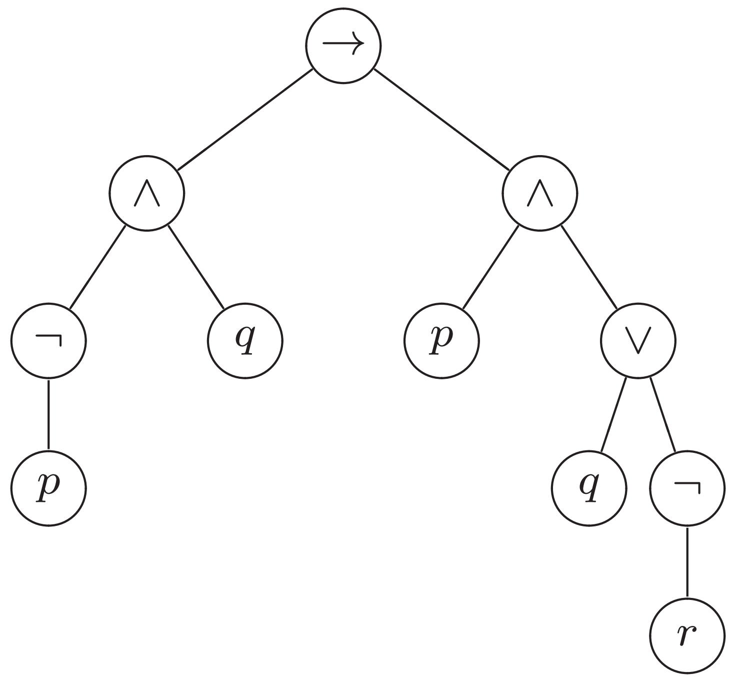
Course-of-values induction:

2. Inductive step: If n is a nonnegative, integer number for which *we assume* that $M(1) \wedge M(2) \wedge \cdots \wedge M(n)$ holds, then *we can show* that $n+1$ has property $M(n+1)$; i.e., we have a proof of $M(1) \wedge M(2) \wedge \cdots \wedge M(n) \rightarrow M(n+1)$.

Structural induction: induction on the structure

Formulas, trees, . . .

$$(((\neg p) \wedge q) \to (p \wedge (q \vee (\neg r))))$$



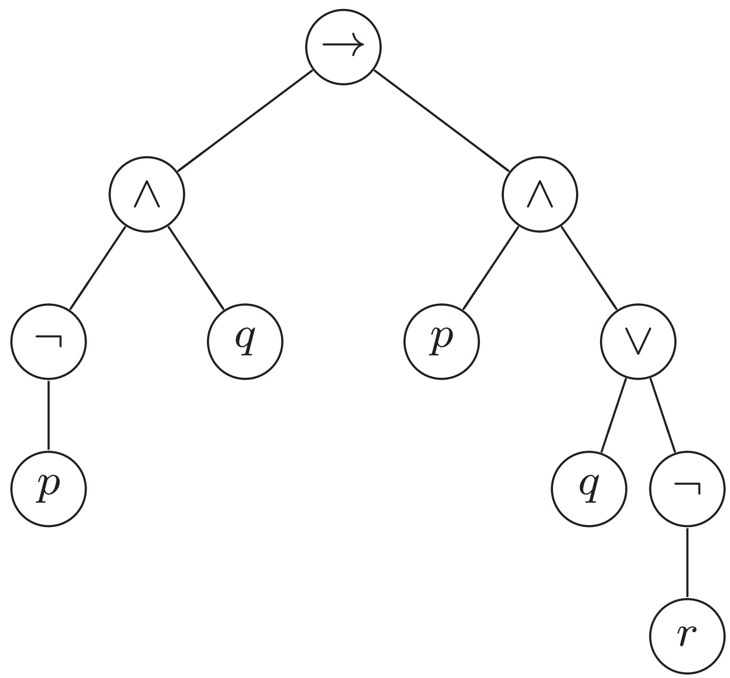
Definition 1.32. Given a well-formed formula ϕ, we define its height to be 1 plus the length of the longest path of its parse tree.

Brackets in a well-formed formula

Theorem 1.33.
For every well-formed propositional logic formula, the number of left brackets is equal to the number of right brackets.

Proof...

$$(((\neg p) \wedge q) \rightarrow (p \wedge (q \vee (\neg r))))$$



Mathematical induction would not work. . .

1.2. Natural deduction

Proof rules

Premises $\phi_1, \phi_2, \ldots, \phi_n$

Conclusion ψ

Sequent $\phi_1, \phi_2, \ldots, \phi_n \vdash \psi$

A slide from lecture 1:

Propositional logic

Example 1.1. If the train arrives late and there are no taxis at the station, then John is late for his meeting. John is not late for his meeting. The train did arrive late.
Therefore, there were taxis at the station.

Example 1.2. If it is raining and Jane does not have her umbrella with her, then she will get wet. Jane is not wet. It is raining.
Therefore, Jane has her umbrella with her.

General structure:
If p and not q, then r. Not r. p. *Therefore,* q.

Propositional logic

Example 1.1. If the train arrives late and there are no taxis at the station, then John is late for his meeting. John is not late for his meeting. The train did arrive late.
Therefore, there were taxis at the station.

Example 1.2. If it is raining and Jane does not have her umbrella with her, then she will get wet. Jane is not wet. It is raining.
Therefore, Jane has her umbrella with her.

General structure:
$p \wedge \neg q \to r, \neg r, p \vdash q$

The rules for conjunction

And-introduction:

$$\frac{\phi \qquad \psi}{\phi \wedge \psi} \wedge \mathsf{i}$$

The rules for conjunction

And-elimination:

$$\frac{\phi \wedge \psi}{\phi} \wedge e_1 \qquad \frac{\phi \wedge \psi}{\psi} \wedge e_2$$

Example 1.4. Proof of: $p \wedge q, r \vdash q \wedge r$

Example 1.4. Proof of: $p \wedge q, r \vdash q \wedge r$

1	$p \wedge q$	premise
2	r	premise
3	q	$\wedge$e$_2$ 1
4	$q \wedge r$	$\wedge$i $3, 2$

In tree-like form. . .

Example 1.6. Proof of: $(p \wedge q) \wedge r, s \wedge t \vdash q \wedge s$