Introduction to logic

Frédéric Prost (frederic.prost@univ-grenoble-alpes.fr)

Handout by Stéphane Devismes   Pascal Lafourcade   Michel Lévy
Benjamin Wack

Université Grenoble Alpes

January 2023

Organization

12 weeks:

- Lecture, 1h30 / week
- Seminar 2 $\times$ 1h30 $=$ 3h / week – Maxime Lesourd (maxime.lesourd@univ-grenoble-alpes.fr)

This week

- Lecture : today (2) ! and Wednesday at 9:45 as usual
- Seminar : Monday at 17:00, Thursday at 09:45

https://lig-membres.imag.fr/prost/INF432/

French version:

https://wackb.gricad-pages.univ-grenoble-alpes.fr/inf402/

Final grade (provided exams can take place...)

Evaluations

- Assessments **60%**:
  4 periodic tests **10%**, midterm exam **20%** and project **30%**
- Exam: **40%**

Project groups: 3-4 students per project group.

- Part 1: Modeling of a logic problem (automated in a software)
- Part 2: Transforming instances of these problems in clauses and solving them using an SAT solver
- Optional: Coding of your own SAT-solver

Examples of problems: N queens, Sudoku-like grids...

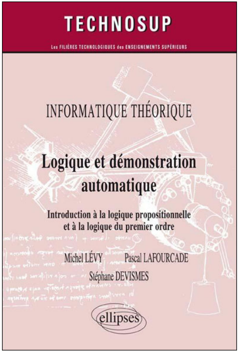
Planning

Important dates

- Project pre-report: March 11th
- Midterm exam: March 14th - 18th
- Project report: April 29th
- Project defense: May 2nd - 13th
- Final exam: May 16th - 25th

Course Material

- Lectures handout (in French) + online pdf
- Subject of the project + online resources



Summary

Summary

Logic

Definitions

- Logic is used to specify what a correct reasoning is, regardless of the application domain.

Logic

Definitions

- Logic is used to specify what a correct reasoning is, regardless of the application domain.
- A reasoning is a way to obtain a conclusion starting from given hypotheses.

Logic

Definitions

- Logic is used to specify what a correct reasoning is, regardless of the application domain.
- A reasoning is a way to obtain a conclusion starting from given hypotheses.
- A correct reasoning does not say anything about the truth of the hypotheses, it only says that **starting from the truth of the hypotheses, one can deduct the truth of the conclusion**.

Examples

Example I

- **Hypothesis I:** All men are mortal
- **Hypothesis II:** Socrates is a man
- **Conclusion:** Socrates is mortal

Examples

Example I

- **Hypothesis I:** All men are mortal
- **Hypothesis II:** Socrates is a man
- **Conclusion:** Socrates is mortal

Example II

- **Hypothesis I:** All that is rare is expensive
- **Hypothesis II:** A cheap horse is rare
- **Conclusion:** A cheap horse is expensive!

Adding a hypothesis

Adding a hypothesis

Example III

- **Hypothesis I:** All that is rare is expensive
- **Hypothesis II:** A cheap horse is rare
- **Hypothesis III:** Every cheap thing is “not expensive”

Adding a hypothesis

Example III

- **Hypothesis I:** All that is rare is expensive
- **Hypothesis II:** A cheap horse is rare
- **Hypothesis III:** Every cheap thing is “not expensive”
- **Conclusion:** Contradictory hypotheses! Since:
  - **Hypothesis I + Hypothesis II:** A cheap horse is expensive
  - **Hypothesis III:** A cheap horse is not expensive

Applications

- **Hardware**: logical gates
- **Software verification and correctness, security**:
  - Tools: provers COQ, PVS, Prover9, MACE, ...
  - Meteor, Airbus...
- **Artificial Intelligence** :
  - expert system (*MyCin*), ontology
- **Programming**: Prolog
  - artificial intelligence
  - natural language processing
- **Certified mathematical proofs**

Course Objectives

- Modeling and formalizing a problem.

Course Objectives

- Modeling and formalizing a problem.
- Understanding a formal reasoning, in particular, being able to determine if it is correct or not.

Course Objectives

- Modeling and formalizing a problem.
- Understanding a formal reasoning, in particular, being able to determine if it is correct or not.
- Reasoning, that is, building a correct reasoning using the tools of propositional logic and first order logic.

Course Objectives

- Modeling and formalizing a problem.
- Understanding a formal reasoning, in particular, being able to determine if it is correct or not.
- Reasoning, that is, building a correct reasoning using the tools of propositional logic and first order logic.
- Writing a rigorous proof, in particular an induction.

Overview of the Semester

TODAY

- Propositional logic
- Propositional resolution
- Natural deduction for propositional logic

MIDTERM EXAM

- First order logic
- Logical basis for automated proving (“first-order resolution”)
- First-order natural deduction

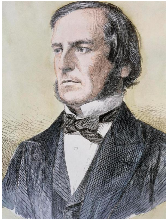
EXAM

Summary

About formal logics

Leibniz (and even the Stoics) already had the idea: in order to reason systematically, you need to formalize: first the statements, then the reasoning itself.

George Boole's approach (1854) is all about computation:



- translate statements into algebraic expressions
- apply computation rules which model the human reasoning
- interpret back the result

Boole's proposal is the starting point of mathematical logic, and propositional logic in particular.

Propositional Logic

Definition

Propositional logic is the logic *without quantifiers*.
The only logical operations used are:

- $\neg$ (negation),
- $\wedge$ (conjunction, also known as logical “and”),
- $\vee$ (disjunction, also known as logical “or”),
- $\Rightarrow$ (implication)
- $\Leftrightarrow$ (equivalence)

Example: **Formal reasoning**

Hypotheses :

- (H1): If Peter is old, then John is not the son of Peter
- (H2): If Peter is not old, then John is the son of Peter
- (H3): If John is Peter's son then Mary is the sister of John

Conclusion (C): Mary is the sister of John, or Peter is old.

Example: **Formal reasoning**

Hypotheses :

- (H1): If Peter is old, then John is not the son of Peter
- (H2): If Peter is not old, then John is the son of Peter
- (H3): If John is Peter's son then Mary is the sister of John

Conclusion (C): Mary is the sister of John, or Peter is old.

- p: "Peter is old"
- j: "John is the son of Peter"
- m: "Mary is the sister of John"

Example: **Formal reasoning**

Hypotheses :

- (H1): If Peter is old, then John is not the son of Peter
- (H2): If Peter is not old, then John is the son of Peter
- (H3): If John is Peter's son then Mary is the sister of John

Conclusion (C): Mary is the sister of John, or Peter is old.

- p: "Peter is old"
- j: "John is the son of Peter"
- m: "Mary is the sister of John"

- (H1): $p \Rightarrow \neg j$
- (H2): $\neg p \Rightarrow j$
- (H3): $j \Rightarrow m$
- (C) : $m \vee p$

Example: **Formal reasoning**

Hypotheses :

- (H1): If Peter is old, then John is not the son of Peter
- (H2): If Peter is not old, then John is the son of Peter
- (H3): If John is Peter's son then Mary is the sister of John

Conclusion (C): Mary is the sister of John, or Peter is old.

- p: "Peter is old"
- j: "John is the son of Peter"
- m: "Mary is the sister of John"

- (H1): $p \Rightarrow \neg j$
- (H2): $\neg p \Rightarrow j$
- (H3): $j \Rightarrow m$
- (C) : $m \vee p$

We prove that $H1 \wedge H2 \wedge H3 \Rightarrow C$:

$$(p \Rightarrow \neg j) \wedge (\neg p \Rightarrow j) \wedge (j \Rightarrow m) \Rightarrow m \vee p$$

is true regardless of the truth value of propositions p, j, m.

Summary

Vocabulary of the language

- The constants: $\top$ (*true*) and $\perp$ (*false*)
- The variables: for example, x, y_1
- The parentheses
- The connectives: $\neg, \vee, \wedge, \Rightarrow, \Leftrightarrow$

(Strict) Formula

Definition 1.1.1

A strict formula is defined inductively as:

- $\top$ and $\perp$ are strict formulae.
- A variable is a strict formula.
- If A is a strict formula then $\neg A$ is a strict formula.
- If A and B are strict formulae and if $\circ$ is one of the following operations $\vee, \wedge, \Rightarrow, \Leftrightarrow$ then $(A \circ B)$ is a strict formula.

(Strict) Formula

Definition 1.1.1

A strict formula is defined inductively as:

- $\top$ and $\bot$ are strict formulae.
- A variable is a strict formula.
- If A is a strict formula then $\neg A$ is a strict formula.
- If A and B are strict formulae and if $\circ$ is one of the following operations $\vee, \wedge, \Rightarrow, \Leftrightarrow$ then $(A \circ B)$ is a strict formula.

Example 1.1.2

$(a \vee (\neg b \wedge c))$ is a strict formula, but not $a \vee (\neg b \wedge c)$, nor $(a \vee (\neg(b) \wedge c))$.

Tree

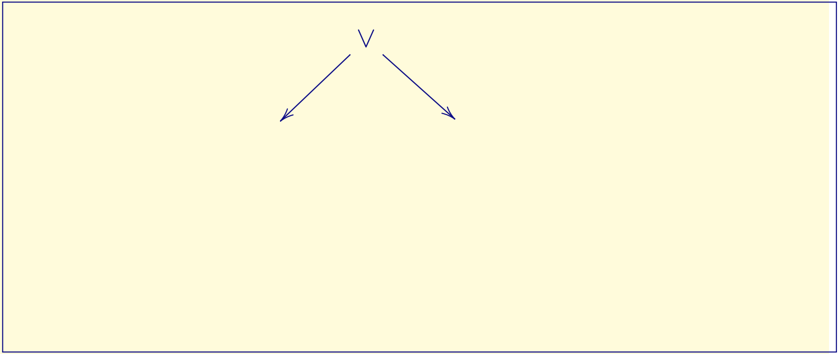
Example 1.1.3

The structure of the formula $(a \vee (\neg b \wedge c))$ is illustrated by the following tree:

Tree

Example 1.1.3

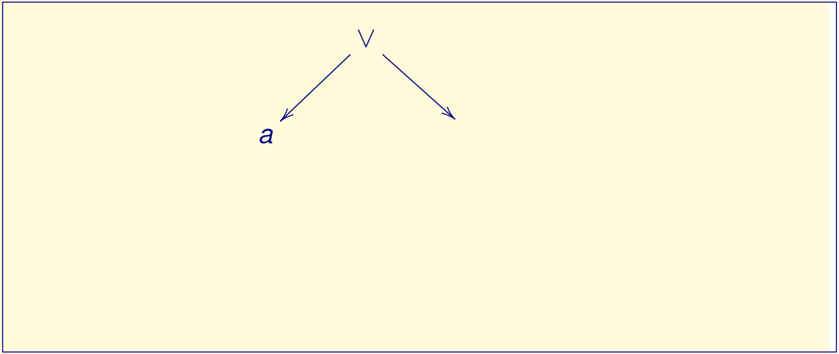
The structure of the formula $(a \vee (\neg b \wedge c))$ is illustrated by the following tree:



Tree

Example 1.1.3

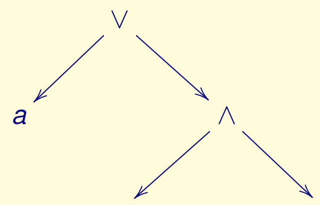
The structure of the formula $(a \vee (\neg b \wedge c))$ is illustrated by the following tree:



Tree

Example 1.1.3

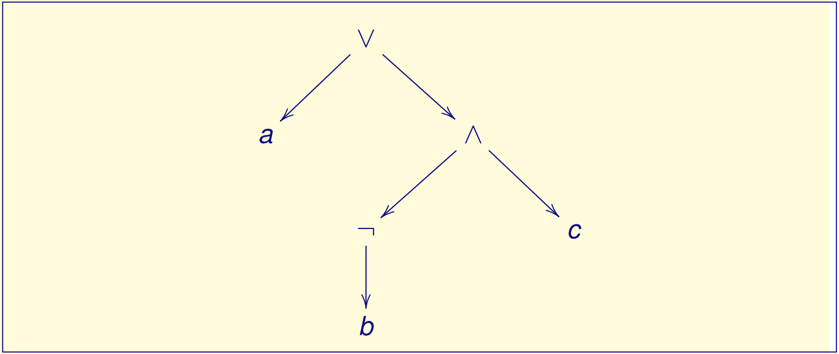
The structure of the formula $(a \vee (\neg b \wedge c))$ is illustrated by the following tree:



Tree

Example 1.1.3

The structure of the formula $(a \vee (\neg b \wedge c))$ is illustrated by the following tree:

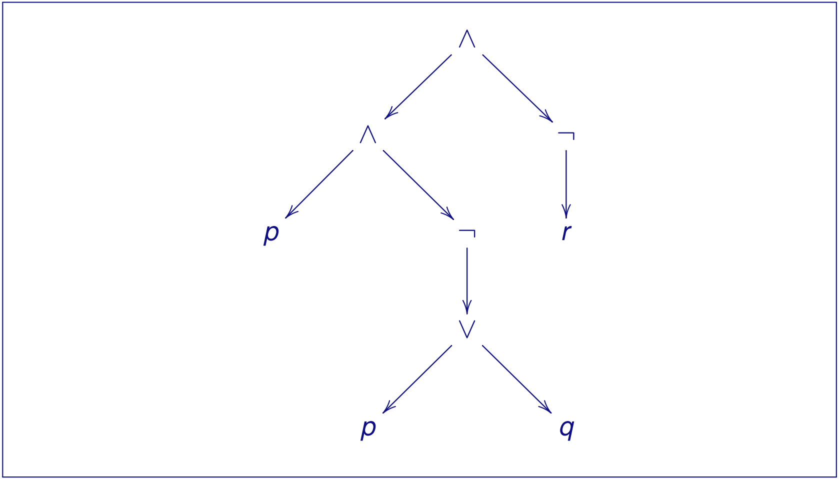


Exercise

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

Exercise

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



Size of a formula

Definition 1.1.10

The size of a formula A, denoted $|A|$, is inductively defined as:

- $|\top| = 0$ and $|\bot| = 0$.
- If A is a variable then $|A| = 0$.
- $|\neg A| = 1 + |A|$.
- $|(A \circ B)| = |A| + |B| + 1$.

Size of a formula

Definition 1.1.10

The size of a formula A, denoted $|A|$, is inductively defined as:

- $|\top| = 0$ and $|\bot| = 0$.
- If A is a variable then $|A| = 0$.
- $|\neg A| = 1 + |A|$.
- $|(A \circ B)| = |A| + |B| + 1$.

Example 1.1.11

$|(a \vee (\neg b \wedge c))| =$

Size of a formula

Definition 1.1.10

The size of a formula A, denoted $|A|$, is inductively defined as:

- $|\top| = 0$ and $|\bot| = 0$.
- If A is a variable then $|A| = 0$.
- $|\neg A| = 1 + |A|$.
- $|(A \circ B)| = |A| + |B| + 1$.

Example 1.1.11

$|(a \vee (\neg b \wedge c))| =$

3.

First result

Strict formulae decompose uniquely in their sub-formulae.

Theorem 1.1.13

For every formula A, there is one and only one of the following cases:

- A is a variable,
- A is a constant,
- A can be written in a unique manner as $\neg B$ where B is a formula,
- A can be written in a unique manner as $(B \circ C)$ where B and C are formulae.

This will allow us to:

- prove properties *by cases*
- perform *structural induction* on the formulae rather than induction on their size.

Prioritized formula

Definition 1.1.14

A prioritized formula is inductively defined in a similar way but:

- if A and B are prioritized formulae the $A \circ B$ is a prioritized formula,
- if A is a prioritized formula then (A) is a prioritized formula.

Example 1.1.15

$a \vee \neg b \wedge c$ is a prioritized formula, but not a (strict) formula.

Connective precedence

Definition 1.1.16

By decreasing precedence, the connectives are: $\neg$, $\wedge$, $\vee$, $\Rightarrow$ and $\Leftrightarrow$.

Left associativity

For identical connectives, the left-hand side connective has higher precedence:
$A \circ B \circ C \;=\; (A \circ B) \circ C$
except for the implication: $A \Rightarrow B \Rightarrow C \;=\; A \Rightarrow (B \Rightarrow C)$

Example of prioritized formulae

Example 1.1.17

- $a \wedge b \wedge c$ is the abbreviation of
- $a \wedge b \vee c$ is the abbreviation of
- $a \vee b \wedge c$ is the abbreviation of

Example of prioritized formulae

Example 1.1.17

- $a \wedge b \wedge c$ is the abbreviation of

  $((a \wedge b) \wedge c)$

- $a \wedge b \vee c$ is the abbreviation of

  $((a \wedge b) \vee c)$

- $a \vee b \wedge c$ is the abbreviation of

  $(a \vee (b \wedge c))$

Summary

Truth assignment of a formula

Definition 1.2.1

A truth assignment is a function
from the set of variables of a formula to the set $\{0, 1\}$.
$[A]_v$ denotes the truth value of the formula A for the assignment v.

Truth assignment of a formula

Definition 1.2.1

A truth assignment is a function
from the set of variables of a formula to the set $\{0, 1\}$.
$[A]_v$ denotes the truth value of the formula A for the assignment v.

Example: Let v be an assignment such that $v(x) = 0$ and $v(y) = 1$.
Applying v to $x \vee y$ is written as

Truth assignment of a formula

Definition 1.2.1

A truth assignment is a function
from the set of variables of a formula to the set $\{0, 1\}$.
$[A]_v$ denotes the truth value of the formula A for the assignment v.

Example: Let v be an assignment such that $v(x) = 0$ and $v(y) = 1$.
Applying v to $x \vee y$ is written as $[x \vee y]_v$
$[x \vee y]_v =$

Truth assignment of a formula

Definition 1.2.1

A truth assignment is a function
from the set of variables of a formula to the set $\{0, 1\}$.
$[A]_v$ denotes the truth value of the formula A for the assignment v.

Example: Let v be an assignment such that $v(x) = 0$ and $v(y) = 1$.
Applying v to $x \vee y$ is written as $[x \vee y]_v$
$[x \vee y]_v = 0 \vee 1 = 1$
Conclusion:

Truth assignment of a formula

Definition 1.2.1

A truth assignment is a function
from the set of variables of a formula to the set $\{0, 1\}$.
$[A]_v$ denotes the truth value of the formula A for the assignment v.

Example: Let v be an assignment such that $v(x) = 0$ and $v(y) = 1$.
Applying v to $x \vee y$ is written as $[x \vee y]_v$
$[x \vee y]_v = 0 \vee 1 = 1$
Conclusion: $x \vee y$ is true for the truth assignment v

Truth value of a formula

Definition 1.2.2

Let A, B be two formulae, x a variable and v a truth assignment.

- $[x]_v =$
- $[\top]_v =$, $[\bot]_v =$
- $[\neg A]_v =$
- $[(A \vee B)]_v =$
- $[(A \wedge B)]_v =$
- $[(A \Rightarrow B)]_v =$
- $[(A \Leftrightarrow B)]_v =$

Truth value of a formula

Definition 1.2.2

Let A, B be two formulae, x a variable and v a truth assignment.

- $[x]_v = v(x)$
- $[\top]_v =$, $[\bot]_v =$
- $[\neg A]_v =$
- $[(A \vee B)]_v =$
- $[(A \wedge B)]_v =$
- $[(A \Rightarrow B)]_v =$
- $[(A \Leftrightarrow B)]_v =$

Truth value of a formula

Definition 1.2.2

Let A, B be two formulae, x a variable and v a truth assignment.

- $[x]_v = v(x)$
- $[\top]_v = 1$, $[\bot]_v =$
- $[\neg A]_v =$
- $[(A \vee B)]_v =$
- $[(A \wedge B)]_v =$
- $[(A \Rightarrow B)]_v =$
- $[(A \Leftrightarrow B)]_v =$

Truth value of a formula

Definition 1.2.2

Let A, B be two formulae, x a variable and v a truth assignment.

- $[x]_v = v(x)$
- $[\top]_v = 1$, $[\bot]_v = 0$
- $[\neg A]_v =$
- $[(A \vee B)]_v =$
- $[(A \wedge B)]_v =$
- $[(A \Rightarrow B)]_v =$
- $[(A \Leftrightarrow B)]_v =$

Truth value of a formula

Definition 1.2.2

Let A, B be two formulae, x a variable and v a truth assignment.

- $[x]_v = v(x)$
- $[\top]_v = 1$, $[\bot]_v = 0$
- $[\neg A]_v = 1 - [A]_v$
- $[(A \vee B)]_v =$
- $[(A \wedge B)]_v =$
- $[(A \Rightarrow B)]_v =$
- $[(A \Leftrightarrow B)]_v =$

Truth value of a formula

Definition 1.2.2

Let A, B be two formulae, x a variable and v a truth assignment.

- $[x]_v = v(x)$
- $[\top]_v = 1$, $[\bot]_v = 0$
- $[\neg A]_v = 1 - [A]_v$
- $[(A \vee B)]_v = max\{[A]_v, [B]_v\}$
- $[(A \wedge B)]_v =$
- $[(A \Rightarrow B)]_v =$
- $[(A \Leftrightarrow B)]_v =$

Truth value of a formula

Definition 1.2.2

Let A, B be two formulae, x a variable and v a truth assignment.

- $[x]_v = v(x)$
- $[\top]_v = 1$, $[\bot]_v = 0$
- $[\neg A]_v = 1 - [A]_v$
- $[(A \vee B)]_v = max\{[A]_v, [B]_v\}$
- $[(A \wedge B)]_v = min\{[A]_v, [B]_v\}$
- $[(A \Rightarrow B)]_v =$
- $[(A \Leftrightarrow B)]_v =$

Truth value of a formula

Definition 1.2.2

Let A, B be two formulae, x a variable and v a truth assignment.

- $[x]_v = v(x)$
- $[\top]_v = 1$, $[\bot]_v = 0$
- $[\neg A]_v = 1 - [A]_v$
- $[(A \vee B)]_v = max\{[A]_v, [B]_v\}$
- $[(A \wedge B)]_v = min\{[A]_v, [B]_v\}$
- $[(A \Rightarrow B)]_v =$ if $[A]_v = 0$ then 1 else $[B]_v$
- $[(A \Leftrightarrow B)]_v =$

Truth value of a formula

Definition 1.2.2

Let A, B be two formulae, x a variable and v a truth assignment.

- $[x]_v = v(x)$
- $[\top]_v = 1$, $[\bot]_v = 0$
- $[\neg A]_v = 1 - [A]_v$
- $[(A \vee B)]_v = max\{[A]_v, [B]_v\}$
- $[(A \wedge B)]_v = min\{[A]_v, [B]_v\}$
- $[(A \Rightarrow B)]_v =$ if $[A]_v = 0$ then 1 else $[B]_v$
- $[(A \Leftrightarrow B)]_v =$ if $[A]_v = [B]_v$ then 1 else 0

Truth table

Definition 1.2.3

A truth table of a formula A is a table representing the truth values of A for **all** the possible values of the variables of A.

- a line of the truth table = an assignment
- a column of the truth table = the truth values of a formula.

Basic tables

0 indicates false and 1 indicates true.
The value of the constant $\top$ is 1 and the value of the constant $\perp$ is 0

Table 1.1 (truth table of connectives)

x	y	$\neg x$	$x \vee y$	$x \wedge y$	$x \Rightarrow y$	$x \Leftrightarrow y$
0	0	1	0	0	1	1
0	1	1	1	0	1	0
1	0	0	1	0	0	0
1	1	0	1	1	1	1

Example:

Example 1.2.4

Give the truth table of the following formulae.

x	y	$x \Rightarrow y$	$\neg x$	$\neg x \vee y$	$(x \Rightarrow y) \Leftrightarrow (\neg x \vee y)$	$x \vee \neg y$
0	0					
0	1					
1	0					
1	1					

Example:

Example 1.2.4

Give the truth table of the following formulae.

x	y	$x \Rightarrow y$	$\neg x$	$\neg x \vee y$	$(x \Rightarrow y) \Leftrightarrow (\neg x \vee y)$	$x \vee \neg y$
0	0	1				
0	1	1				
1	0	0				
1	1	1				

Example:

Example 1.2.4

Give the truth table of the following formulae.

x	y	$x \Rightarrow y$	$\neg x$	$\neg x \vee y$	$(x \Rightarrow y) \Leftrightarrow (\neg x \vee y)$	$x \vee \neg y$
0	0	1	1			
0	1	1	1			
1	0	0	0			
1	1	1	0			

Example:

Example 1.2.4

Give the truth table of the following formulae.

x	y	$x \Rightarrow y$	$\neg x$	$\neg x \vee y$	$(x \Rightarrow y) \Leftrightarrow (\neg x \vee y)$	$x \vee \neg y$
0	0	1	1	1		
0	1	1	1	1		
1	0	0	0	0		
1	1	1	0	1		

Example:

Example 1.2.4

Give the truth table of the following formulae.

x	y	$x \Rightarrow y$	$\neg x$	$\neg x \vee y$	$(x \Rightarrow y) \Leftrightarrow (\neg x \vee y)$	$x \vee \neg y$
0	0	1	1	1	1	
0	1	1	1	1	1	
1	0	0	0	0	1	
1	1	1	0	1	1	

Example:

Example 1.2.4

Give the truth table of the following formulae.

x	y	$x \Rightarrow y$	$\neg x$	$\neg x \vee y$	$(x \Rightarrow y) \Leftrightarrow (\neg x \vee y)$	$x \vee \neg y$
0	0	1	1	1	1	1
0	1	1	1	1	1	0
1	0	0	0	0	1	1
1	1	1	0	1	1	1

Equivalent formulae

Definition 1.2.5

Two formulae A and B are equivalent (denoted $A \equiv B$ or simply $A = B$) if they have the same truth value for every assignment.

Equivalent formulae

Definition 1.2.5

Two formulae A and B are equivalent (denoted $A \equiv B$ or simply $A = B$) if they have the same truth value for every assignment.

Example 1.2.6

$x \Rightarrow y \equiv \neg x \vee y$

Equivalent formulae

Definition 1.2.5

Two formulae A and B are equivalent (denoted $A \equiv B$ or simply $A = B$) if they have the same truth value for every assignment.

Example 1.2.6

$x \Rightarrow y \equiv \neg x \vee y$

Remark:
The logical connective $\Leftrightarrow$ does not mean $A \equiv B$.

Validity, tautology (1/2)

Definition 1.2.8

- A formula is valid if its value is 1 for all truth assignments.
- A valid formula is also called a tautology.
- Denoted by $\models A$.

Validity, tautology (1/2)

Definition 1.2.8

- A formula is valid if its value is 1 for all truth assignments.
- A valid formula is also called a tautology.
- Denoted by $\models A$.

Example 1.2.9

- $(x \Rightarrow y) \Leftrightarrow (\neg x \vee y)$ is valid;
- $x \Rightarrow y$ is not valid since

  it is false for $x = 1$ and $y = 0$.

Valid, tautology (2/2)

Property 1.2.10

The formulae A and B are equivalent ($A \equiv B$)
if and only if
formula $A \Leftrightarrow B$ is valid.

Proof.

The property is a consequence of the truth table of $\Leftrightarrow$. □

Model for a formula

Definition 1.2.11

A truth assignment v for which a formula has truth value equal to 1 is a model for that formula.

v satisfies A or v makes A true.

Example 1.2.12

A model for $x \Rightarrow y$ is:

Model for a formula

Definition 1.2.11

A truth assignment v for which a formula has truth value equal to 1 is a model for that formula.

v satisfies A or v makes A true.

Example 1.2.12

A model for $x \Rightarrow y$ is:

$x = 1, y = 1$ (among others)

Model for a formula

Definition 1.2.11

A truth assignment v for which a formula has truth value equal to 1 is a model for that formula.

v satisfies A or v makes A true.

Example 1.2.12

A model for $x \Rightarrow y$ is:

$x = 1, y = 1$ (among others)

On the opposite, $x = 1$, $y = 0$ is not a model for $x \Rightarrow y$.

Model for a set of formulae

Definition 1.2.13

v is a model for a set of formulae $\{A_1, \ldots, A_n\}$
if and only if
it is a model for every formula in the set.

Model for a set of formulae

Definition 1.2.13

v is a model for a set of formulae $\{A_1, \ldots, A_n\}$
if and only if
it is a model for every formula in the set.

Example 1.2.14

A model of $\{a \Rightarrow b, b \Rightarrow c\}$ is:

Model for a set of formulae

Definition 1.2.13

v is a model for a set of formulae $\{A_1, \ldots, A_n\}$
if and only if
it is a model for every formula in the set.

Example 1.2.14

A model of $\{a \Rightarrow b, b \Rightarrow c\}$ is:

$a = 0, b = 0$ (for any c).

Property of a model for a set of formulae

Property 1.2.15

v is a model for $\{A_1, \ldots, A_n\}$
if and only if
v is a model for $A_1 \wedge \ldots \wedge A_n$.

Property of a model for a set of formulae

Property 1.2.15

v is a model for $\{A_1, \ldots, A_n\}$
if and only if
v is a model for $A_1 \wedge \ldots \wedge A_n$.

Example 1.2.16

The set of formulae $\{a \Rightarrow b, b \Rightarrow c\}$
and the formula $(a \Rightarrow b) \wedge (b \Rightarrow c)$
have identical models.

Counter-model

Definition 1.2.17

A truth assignment v which yields the value 0 for a formula is a counter-model for the formula.

v does not satisfy the formula or v makes the formula false.

Counter-model

Definition 1.2.17

A truth assignment v which yields the value 0 for a formula is a counter-model for the formula.

v does not satisfy the formula or v makes the formula false.

Example 1.2.18

A counter-model of $x \Rightarrow y$ is:

Counter-model

Definition 1.2.17

A truth assignment v which yields the value 0 for a formula is a counter-model for the formula.

v does not satisfy the formula or v makes the formula false.

Example 1.2.18

A counter-model of $x \Rightarrow y$ is:

$x = 1, y = 0.$

Satisfiable formula

Definition 1.2.20

A (set of) formula(e) is satisfiable if it admits a model.

Definition 1.2.21

A (set of) formula(e) is unsatisfiable if it is not satisfiable.

Satisfiable formula

Definition 1.2.20

A (set of) formula(e) is satisfiable if it admits a model.

Definition 1.2.21

A (set of) formula(e) is unsatisfiable if it is not satisfiable.

Example 1.2.22

$x \wedge \neg x$ is unsatisfiable, but $x \Rightarrow y$ is satisfiable.

Satisfiable formula

Definition 1.2.20

A (set of) formula(e) is satisfiable if it admits a model.

Definition 1.2.21

A (set of) formula(e) is unsatisfiable if it is not satisfiable.

Example 1.2.22

$x \wedge \neg x$ is unsatisfiable, but $x \Rightarrow y$ is satisfiable.

Beware

unsatisfiable = 0 model	satisfiable = at least 1 model
invalid = at least 1 counter-model	valid = 0 counter-model

Summary

Today

- Why define and use formal logic?
- Propositional logic:
  - 1 variable = 1 proposition (a statement) which may be true or false
  - 5 connectives to articulate these propositions
- Meaning of formulae :
  - assignment = choice of a truth value for each variable
  - a formula may be true for 0, 1, several or every assignement

Next time

Homework: build the truth table for the “Peter, John and Mary” example.

- Important equivalences
- Substitutions and replacements
- Normal Forms

Why study?

The more I study, the more I know
The more I know, the more I forget
The more I forget, the less I know