Previous lecture

- Why formal logic?
- Propositional logic
- Syntax
- Meaning of formulae
Our example with a truth table

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.

\[(p \Rightarrow \neg j) \land (\neg p \Rightarrow j) \land (j \Rightarrow m) \Rightarrow m \lor p\]
Our example with a truth table

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.

\[(p \Rightarrow \neg j) \land (\neg p \Rightarrow j) \land (j \Rightarrow m) \Rightarrow m \lor p\]

<p>| | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>p</td>
<td>j</td>
<td>m</td>
<td>p \Rightarrow \neg j</td>
<td>\neg p \Rightarrow j</td>
<td>j \Rightarrow m</td>
<td>H₁ \land H₂ \land H₃</td>
<td>m \lor p</td>
<td>H₁ \land H₂ \land H₃ \Rightarrow m \lor p</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</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>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
Plan

Consequence

Important equivalences

Substitution and replacement

Normal forms

Conclusion
Plan

Consequence

Important equivalences

Substitution and replacement

Normal forms

Conclusion
**Logical consequence (entailment)**

**Definition 1.2.24**

$A$ is a **consequence** of the set $\Gamma$ of hypotheses ($\Gamma \models A$) if every model of $\Gamma$ is a model of $A$.

**Remark 1.2.26**

$\models A$ denotes that $A$ is valid.

(Every truth assignment is a model for the empty set.)
Example of a consequence

Example 1.2.28

\[ a \rightarrow b, b \rightarrow c \models a \rightarrow c. \]
Example of a consequence

Example 1.2.28

\[ a \implies b, b \implies c \models a \implies c. \]

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>b</td>
<td>c</td>
<td>a \implies b</td>
<td>b \implies c</td>
<td>a \implies c</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<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>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<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>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
ESSENTIAL property

Often used in exercises and during exams.

Property 1.2.27

Let \( H_n = A_1 \land \ldots \land A_n \).

The following three formulations are equivalent:

1. \( A_1, \ldots, A_n \models B \)
2. \( H_n \Rightarrow B \) is valid.
3. \( H_n \land \neg B \) is unsatisfiable.

Proof.

Based on the truth tables of the connectives.
We prove that 1 \( \Rightarrow \) 2 then 2 \( \Rightarrow \) 3 and 3 \( \Rightarrow \) 1.
Proof (1/3)

1 ⇒ 2: let us assume that $A_1, \ldots, A_n \models B$.

Let $\nu$ be a truth assignment:

- if $\nu$ is not a model for $A_1, \ldots, A_n$:
  - for a certain $i$ we have $[A_i]_\nu = 0$, hence $[H_n]_\nu = 0$.
  - Thus $[H_n \Rightarrow B]_\nu = 1$.

- if $\nu$ is a model for $A_1, \ldots, A_n$:
  - then by hypothesis $\nu$ is a model for $B$ therefore $[B]_\nu = 1$.
  - Thus $[H_n \Rightarrow B]_\nu = 1$.

Therefore $H_n \Rightarrow B$ is valid.
Proof (2/3)

2 ⇒ 3: let us assume that \( H_n \Rightarrow B \) is valid.
For every truth assignment \( v \):
- either \([H_n]_v = 0\),
- or \([H_n]_v = 1\) and \([B]_v = 1\).

However \([H_n \land \neg B]_v = \min([H_n]_v, [\neg B]_v) = \min([H_n]_v, 1 - [B]_v)\).
In both cases, we have \([H_n \land \neg B]_v = 0\).
Therefore \( H_n \land \neg B \) is unsatisfiable.
**Proof (3/3)**

1. Let us assume that $H_n \land \neg B$ is unsatisfiable. Let us show that $A_1, \ldots, A_n \models B$.

Let $\nu$ be a truth assignment model of $A_1, \ldots, A_n$:

- $[H_n]_{\nu} = [A_1 \land \ldots \land A_n]_{\nu} = 1$.
- According to our hypothesis $[-B]_{\nu} = 0$.

Hence, $1 - [B]_{\nu} = 0$ so $[B]_{\nu} = 1$, i.e. $\nu$ is a model for $B$.

Exercise 7 shows why proving these 3 circular implications is sufficient.
### Instance of the property

**Example 1.2.28**

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th>a ⇒ b</th>
<th>b ⇒ c</th>
<th>a ⇒ c</th>
<th>(a ⇒ b) ∧ (b ⇒ c) ⇒ (a ⇒ c)</th>
<th>(a ⇒ b) ∧ (b ⇒ c) ∧ ¬(a ⇒ c)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Instance of the property

Example 1.2.28

<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
<th>c</th>
<th>( a \Rightarrow b )</th>
<th>( b \Rightarrow c )</th>
<th>( a \Rightarrow c )</th>
<th>( (a \Rightarrow b) \land (b \Rightarrow c) ) ( \Rightarrow (a \Rightarrow c) )</th>
<th>( (a \Rightarrow b) \land (b \Rightarrow c) ) ( \land \neg (a \Rightarrow c) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
Instance of the property

Example 1.2.28

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th>( a \Rightarrow b )</th>
<th>b ( \Rightarrow c )</th>
<th>a ( \Rightarrow c )</th>
<th>(( a \Rightarrow b ) ( \land ) (b ( \Rightarrow c )) ( \Rightarrow (a \Rightarrow c) ))</th>
<th>(( a \Rightarrow b ) ( \land ) (b ( \Rightarrow c )) ( \land \lnot (a \Rightarrow c) ))</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>
Compactness

Theorem 1.2.30 Propositional compactness

A set of *propositional* formulae has a model if and only if every finite subset of it has a model.
Compactness

Theorem 1.2.30 Propositional compactness

A set of propositional formulae has a model if and only if every finite subset of it has a model.

This theorem may look trivial. However, the set of formulae may be infinite!

This result will be used at a later stage in the course (bases for automated theorem proving).
Plan

Consequence

Important equivalences

Substitution and replacement

Normal forms

Conclusion
Preamble

How to prove that a formula is valid?
Preamble

How to prove that a formula is valid?

- Truth table
  - Problem: for a formula having 100 variables, the truth table will contain $2^{100}$ lines (unable to be computed, even by a computer!).
Preamble

How to prove that a formula is valid?

- Truth table
  - Problem: for a formula having 100 variables, the truth table will contain $2^{100}$ lines (unable to be computed, even by a computer!).

- Idea:
  - Simplify the formula using transformations
  - Then, study the simplified formula using truth tables or a logic reasoning
Disjunction

- **associative** \( x \lor (y \lor z) \equiv (x \lor y) \lor z \)
- **commutative** \( x \lor y \equiv y \lor x \)
- **idempotent** \( x \lor x \equiv x \)
Disjunction

- **associative** \( x \lor (y \lor z) \equiv (x \lor y) \lor z \)
- **commutative** \( x \lor y \equiv y \lor x \)
- **idempotent** \( x \lor x \equiv x \)

Ditto for conjunction.
Distributivity

- Conjunction distributes over disjunction
  \[ x \land (y \lor z) \equiv (x \land y) \lor (x \land z) \]
Distributivity

- Conjunction distributes over disjunction
  \[ x \land (y \lor z) \equiv (x \land y) \lor (x \land z) \]

- Disjunction distributes over conjunction
  \[ x \lor (y \land z) \equiv (x \lor y) \land (x \lor z) \]
Neutrality and Absorption

- $0$ is the neutral element for disjunction $0 \lor x \equiv x$
- $1$ is the neutral element for conjunction $1 \land x \equiv x$
- $1$ is the absorbing element for disjunction $1 \lor x \equiv 1$
- $0$ is the absorbing element for conjunction $0 \land x \equiv 0$
Negation

**Negation laws:**

1. \( x \land \neg x \equiv 0 \)
2. \( x \lor \neg x \equiv 1 \) (The law of excluded middle)
3. \( \neg \neg x \equiv x \)
4. \( \neg 0 \equiv 1 \)
5. \( \neg 1 \equiv 0 \)
De Morgan laws

\[ \neg (x \land y) \equiv \neg x \lor \neg y \]

\[ \neg (x \lor y) \equiv \neg x \land \neg y \]
Augustus De Morgan (1860) builds on Boole’s algebra:

- Work about quantifiers
- Calculus of relations (also see C.S. Peirce’s works)

which laid grounds for first-order logic (see 2nd part of the course).

- Notion of duality in Boole’s algebras expressed in particular as De Morgan’s laws

- Involved (though very briefly) in the first conjectures about the four colour theorem
Simplification laws

Property 1.2.31

For every $x, y$ we have:

- $x \lor (x \land y) \equiv x$
- $x \land (x \lor y) \equiv x$
- $x \lor (\neg x \land y) \equiv x \lor y$
Plan

Consequence

Important equivalences

Substitution and replacement

Normal forms

Conclusion
Substitution

Definition 1.3.1

A substitution $\sigma$ is a function mapping variables to formulae.
Substitution

Definition 1.3.1

A substitution $\sigma$ is a function mapping variables to formulae.

$A\sigma = \text{the formula } A \text{ where all variables } x \text{ are replaced by the formula } \sigma(x)$. 
Substitution

Definition 1.3.1

A substitution $\sigma$ is a function mapping variables to formulae.

$A\sigma =$ the formula $A$ where all variables $x$ are replaced by the formula $\sigma(x)$.

Example: $A = \neg(p \land q) \iff (\neg p \lor \neg q)$

- Let $\sigma$ the following substitution: $\sigma(p) = (a \lor b), \sigma(q) = (c \land d)$
- $A\sigma =$
Substitution

**Definition 1.3.1**

A substitution $\sigma$ is a function mapping variables to formulae.

$A\sigma$ = the formula $A$ where all variables $x$ are replaced by the formula $\sigma(x)$.

**Example:** $A = \neg(p \land q) \iff (\neg p \lor \neg q)$

- Let $\sigma$ the following substitution: $\sigma(p) = (a \lor b), \sigma(q) = (c \land d)$
- $A\sigma = \neg((a \lor b) \land (c \land d)) \iff (\neg (a \lor b) \lor \neg (c \land d))$
Finite support substitution

Definition 1.3.2

- The support of a substitution $\sigma$ is the set of variables $x$ such that $x \sigma \neq x$.
- A finite support substitution $\sigma$ is denoted $< x_1 := A_1, \ldots, x_n := A_n >$
Finite support substitution

**Definition 1.3.2**

- The support of a substitution $\sigma$ is the set of variables $x$ such that $x \sigma \neq x$.
- A finite support substitution $\sigma$ is denoted by $< x_1 := A_1, \ldots, x_n := A_n >$

**Example 1.3.3**

$A = x \lor x \land y \Rightarrow z \land y$ and $\sigma = < x := a \lor b, z := b \land c >$

$A \sigma =$
Finite support substitution

Definition 1.3.2

- The support of a substitution $\sigma$ is the set of variables $x$ such that $x \sigma \neq x$.
- A finite support substitution $\sigma$ is denoted $< x_1 := A_1, \ldots, x_n := A_n >$

Example 1.3.3

$A = x \lor x \land y \Rightarrow z \land y$ and $\sigma = < x := a \lor b, z := b \land c >$

$A \sigma = (a \lor b) \lor (a \lor b) \land y \Rightarrow (b \land c) \land y$
Properties of substitutions

Property 1.3.4

Let $\nu$ be a truth assignment and $\sigma$ a substitution. Let $w$ be the assignment $w : x \mapsto [\sigma(x)]_\nu$. For any formula $A$, we have $[A\sigma]_\nu = [A]_w$. 

Example 1.3.5:

Let $A = x \lor y \lor d$. Let $\sigma = \langle x := a \lor b, y := b \land c \rangle$. Let $\nu$ be $\nu(a) = 1$, $\nu(b) = 0$, $\nu(c) = 0$, $\nu(d) = 0$. Then $A\sigma = (a \lor b) \lor (b \land c) \lor d$, $[A\sigma]_\nu = (1 \lor 0) \lor (0 \land 0) \lor 0 = 1$, and $[A]_w = 1 \lor 0 \lor 0 = 1$. 

B. Wack et al. (UGA)
Properties of substitutions

Property 1.3.4

Let $\nu$ be a truth assignment and $\sigma$ a substitution. Let $w$ be the assignment $w : x \mapsto [\sigma(x)]_\nu$. For any formula $A$, we have $[A\sigma]_\nu = [A]_w$.

Example 1.3.5:

Let $A = x \lor y \lor d$
Let $\sigma = < x := a \lor b, y := b \land c >$
Let $\nu$ be $\nu(a) = 1, \nu(b) = 0, \nu(c) = 0, \nu(d) = 0$
Properties of substitutions

Property 1.3.4

Let $v$ be a truth assignment and $\sigma$ a substitution. Let $w$ be the assignment $w : x \mapsto [\sigma(x)]_v$. For any formula $A$, we have $[A\sigma]_v = [A]_w$.

Example 1.3.5 :
Let $A = x \lor y \lor d$
Let $\sigma = \langle x := a \lor b, y := b \land c \rangle$
Let $v$ be $v(a) = 1, v(b) = 0, v(c) = 0, v(d) = 0$

\[
A\sigma = (a \lor b) \lor (b \land c) \lor d
\]

\[
[A\sigma]_v = (1 \lor 0) \lor (0 \land 0) \lor 0
= 1 \lor 0 \lor 0 = 1
\]
Properties of substitutions

Property 1.3.4

Let \( v \) be a truth assignment and \( \sigma \) a substitution.
Let \( w \) be the assignment \( w : x \mapsto [\sigma(x)]_v \).
For any formula \( A \), we have \([A\sigma]_v = [A]_w\).

Example 1.3.5:
Let \( A = x \lor y \lor d \)
Let \( \sigma = < x := a \lor b, y := b \land c > \)
Let \( v \) be \( v(a) = 1, v(b) = 0, v(c) = 0, v(d) = 0 \)

\[
A\sigma = (a \lor b) \lor (b \land c) \lor d \\
\begin{align*}
[w(x)]_v &= 1 \lor 0 = 1 \\
[w(y)]_v &= 0 \land 0 = 0 \\
[w(d)]_v &= 0
\end{align*}
\]
\[
[A\sigma]_v = (1 \lor 0) \lor (0 \land 0) \lor 0 \\
= 1 \lor 0 \lor 0 = 1 \\
\]
\[
[A]_w = 1 \lor 0 \lor 0 = 1
\]
Initial step: $|A| = 0$

Two possible cases:

- If $A$ is $\top$ or $\bot$ then $A\sigma = A$ and $[A]_v$ does not depend on $v$.
- If $A$ is a variable $x$, then by construction $[x\sigma]_v = w(x)$. 
Induction

**Hypothesis:** Assume the property holds for any formula of height less or equal to $n$.

Let $A$ be a formula of height $n + 1$; there are two possible cases:

- **Case 1:** $A = \neg B$ with $|B| = n$.

  $\begin{align*}
  [A\sigma]_v &= [\neg B\sigma]_v = [\neg(B\sigma)]_v = 1 - [B\sigma]_v \\
  [A]_w &= [\neg B]_w = 1 - [B]_w.
  \end{align*}$

  Since $|B| = n$, by induction hypothesis $[B\sigma]_v = [B]_w$

  Hence, $[A\sigma]_v = [A]_w$. 
Induction

Hypothesis: Assume the property is true for any formula of height less or equal to $n$.
Let $A$ be a formula of height $n + 1$; there are two possible cases:

- **Case 2:** $A = (B \circ C)$ with $|B| < n + 1$ and $|C| < n + 1$.
  
  Then $[A\sigma]_v = [B\sigma \circ C\sigma]_v$
  and $[A]_w = [B \circ C]_w$

  By induction hypothesis $[B\sigma]_v = [B]_w$ and $[C\sigma]_v = [C]_w$.
  Since the semantics for $\circ$ remain the same, $[A\sigma]_v = [A]_w$. 
Substitution of a valid formula

Theorem 1.3.6

If $A$ is valid then $A\sigma$ is valid for any $\sigma$.

Proof.
Substitution of a valid formula

**Theorem 1.3.6**

If $A$ is valid then $A\sigma$ is valid for any $\sigma$.

**Proof.**

Let $v$ be any truth assignment.
Substitution of a valid formula

Theorem 1.3.6

If \( A \) is valid then \( A\sigma \) is valid for any \( \sigma \).

Proof.

Let \( \nu \) be any truth assignment.

According to property 1.3.4: \( [A\sigma]_\nu = [A]_w \) where \( w(x) = [\sigma(x)]_\nu \).
Substitution of a valid formula

**Theorem 1.3.6**

If $A$ is valid then $A\sigma$ is valid for any $\sigma$.

**Proof.**

Let $\nu$ be any truth assignment.

According to property 1.3.4: $[A\sigma]_{\nu} = [A]_{\nu}$ where $\nu(x) = [\sigma(x)]_{\nu}$.

Since $A$ is valid, $[A]_{\nu} = 1$.

Consequently, $A\sigma$ equals 1 in every truth assignment, therefore $A\sigma$ is a valid formula.
Examples

Example 1.3.7

Let $A$ be the formula $\neg(p \land q) \iff (\neg p \lor \neg q)$. This formula is valid, it is an important equivalence. Let $\sigma$ the following substitution: $< p := (a \lor b), q := (c \land d) >$. The formula
Examples

Example 1.3.7

Let $A$ be the formula $\neg (p \land q) \iff (\neg p \lor \neg q)$. This formula is valid, it is an important equivalence. Let $\sigma$ the following substitution:

$< p := (a \lor b), q := (c \land d) >$. The formula

$A\sigma = \neg ((a \lor b) \land (c \land d)) \iff (\neg (a \lor b) \lor \neg (c \land d))$ is also valid.
Replacement

Definition 1.3.8

The formula \( D \) is obtained by replacing certain occurrences of \( A \) by \( B \) in \( C \) if:

\[ \begin{align*}
\text{\( C \) can be written } E < x := A > \\
\text{\( D \) can be written } E < x := B >
\end{align*} \]

for some formula \( E \).
Examples

Example 1.3.9

Consider the formula $C = ((a \Rightarrow b) \lor \neg (a \Rightarrow b))$.

- The formula obtained by replacing all occurrences of $(a \Rightarrow b)$ by $(a \land b)$ is
Example 1.3.9

Consider the formula $C = ((a \Rightarrow b) \lor \neg(a \Rightarrow b))$.

- The formula obtained by replacing all occurrences of $(a \Rightarrow b)$ by $(a \land b)$ is

  $$D = ((a \land b) \lor \neg(a \land b))$$

using $E = (x \lor \neg x)$.
Examples

Example 1.3.9

Consider the formula \( C = ((a \Rightarrow b) \lor \neg(a \Rightarrow b)) \).

- The formula obtained by replacing all occurrences of \( (a \Rightarrow b) \) by \( (a \land b) \) is

\[
D = ((a \land b) \lor \neg(a \land b))
\]

using \( E = (x \lor \neg x) \).

- The formula obtained by replacing the first occurrence of \( (a \Rightarrow b) \) by \( (a \land b) \) is
Examples

Example 1.3.9

Consider the formula $C = ((a \Rightarrow b) \lor \neg(a \Rightarrow b))$.

- The formula obtained by replacing all occurrences of $(a \Rightarrow b)$ by $(a \land b)$ is
  
  
  $D = ((a \land b) \lor \neg(a \land b))$

  using $E = (x \lor \neg x)$.

- The formula obtained by replacing the first occurrence of $(a \Rightarrow b)$ by $(a \land b)$ is
  
  $D = ((a \land b) \lor \neg(a \Rightarrow b))$

  using $E = (x \lor \neg(a \Rightarrow b))$. 
Properties of the replacements

Theorem 1.3.10

If $D$ is obtained by replacing, in $C$, some occurrences of $A$ by $B$, then $(A \iff B) \models (C \iff D)$. 

Proof.
By definition, $C = E <x := A>$ and $D = E <x := B>$. Assume that $[A]_v = [B]_v$, then $w$ is the same for both substitutions. Therefore $[C]_v = [D]_v$: the assignment $v$ is a model of $(C \iff D)$.

Example 1.3.12:
$p \iff q \models (p \lor (p \Rightarrow r)) \iff (p \lor (q \Rightarrow r))$. 

Corollary 1.3.11
Let $D$ be obtained by replacing, in $C$, one occurrence of $A$ by $B$. If $A \equiv B$ then $C \equiv D$.
Properties of the replacements

Theorem 1.3.10

If $D$ is obtained by replacing, in $C$, some occurrences of $A$ by $B$, then

$$(A \iff B) \models (C \iff D).$$

Proof.

By definition, $C = E < x := A >$ et $D = E < x := B >$.

Assume that $[A]_v = [B]_v$, then $w$ is the same for both substitutions. Therefore $[C]_v = [D]_v :$ the assignment $v$ is a model of $(C \iff D)$.  \qed
Properties of the replacements

Theorem 1.3.10

If $D$ is obtained by replacing, in $C$, some occurrences of $A$ by $B$, then

$$(A \Leftrightarrow B) \models (C \Leftrightarrow D).$$

Proof.

By definition, $C = E < x := A >$ et $D = E < x := B >$.
Assume that $[A]_v = [B]_v$, then $w$ is the same for both substitutions. Therefore $[C]_v = [D]_v$ : the assignment $v$ is a model of $(C \Leftrightarrow D)$. $\Box$

Example 1.3.12: $p \Leftrightarrow q \models (p \lor (p \Rightarrow r)) \Leftrightarrow (p \lor (q \Rightarrow r))$. 
Properties of the replacements

Theorem 1.3.10

If $D$ is obtained by replacing, in $C$, some occurrences of $A$ by $B$, then

$$(A \leftrightarrow B) \models (C \leftrightarrow D).$$

Proof.

By definition, $C = E < x := A >$ et $D = E < x := B >$. Assume that $[A]_v = [B]_v$, then $w$ is the same for both substitutions. Therefore $[C]_v = [D]_v$: the assignment $v$ is a model of $(C \leftrightarrow D)$. 

Example 1.3.12:

$p \leftrightarrow q \models (p \lor (p \Rightarrow r)) \iff (p \lor (q \Rightarrow r))$.

Corollary 1.3.11

Let $D$ be obtained by replacing, in $C$, one occurrence of $A$ by $B$. If $A \equiv B$ then $C \equiv D$. 
Plan

Consequence

Important equivalences

Substitution and replacement

Normal forms

Conclusion
Definitions

Definition 1.4.1

▶ A literal is a variable or its negation.
Definitions

Definition 1.4.1

- A **literal** is a variable or its negation.
- A **monomial** is a conjunction of literals (special cases 0 and 1).
Definitions

Definition 1.4.1

- A **literal** is a variable or its negation.
- A **monomial** is a conjunction of literals (special cases 0 and 1).
- A **clause** is a disjunction of literals (special cases 0 and 1).
Definitions

Definition 1.4.1

- A literal is a variable or its negation.
- A monomial is a conjunction of literals (special cases 0 and 1).
- A clause is a disjunction of literals (special cases 0 and 1).

Example 1.4.2

- $x, y, \neg z$ are literals.
Definitions

Definition 1.4.1

▶ A literal is a variable or its negation.
▶ A monomial is a conjunction of literals (special cases 0 and 1).
▶ A clause is a disjunction of literals (special cases 0 and 1).

Example 1.4.2

▶ $x, y, \neg z$ are literals.
▶ $x \land \neg y \land z$ is a monomial
Definitions

Definition 1.4.1

- A literal is a variable or its negation.
- A monomial is a conjunction of literals (special cases 0 and 1).
- A clause is a disjunction of literals (special cases 0 and 1).

Example 1.4.2

- $x, y, \neg z$ are literals.
- $x \land \neg y \land z$ is a monomial
- The monomial $x \land \neg y \land z \land \neg x$ contains $x$ and $\neg x$: its value is 0.
Definitions

**Definition 1.4.1**

- A **literal** is a variable or its negation.
- A **monomial** is a conjunction of literals (special cases 0 and 1).
- A **clause** is a disjunction of literals (special cases 0 and 1).

**Example 1.4.2**

- $x, y, \neg z$ are literals.
- $x \land \neg y \land z$ is a monomial
- The monomial $x \land \neg y \land z \land \neg x$ contains $x$ and $\neg x$: its value is 0.
- $x \lor \neg y \lor z$ is a clause
Definitions

Definition 1.4.1
- A literal is a variable or its negation.
- A monomial is a conjunction of literals (special cases 0 and 1).
- A clause is a disjunction of literals (special cases 0 and 1).

Example 1.4.2
- $x, y, \neg z$ are literals.
- $x \land \neg y \land z$ is a monomial
- The monomial $x \land \neg y \land z \land \neg x$ contains $x$ and $\neg x$: its value is 0.
- $x \lor \neg y \lor z$ is a clause
- The clause $x \lor \neg y \lor z \lor \neg x$ contains $x$ and $\neg x$: its value is 1.
### Normal form

<table>
<thead>
<tr>
<th>Definition 1.4.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>A formula is in <strong>normal form</strong> if it only contains the operators $\land, \lor, \neg$ and the negations are only applied to <strong>variables</strong>.</td>
</tr>
</tbody>
</table>
Normal form

Definition 1.4.3

A formula is in **normal form** if it only contains the operators $\land, \lor, \lnot$ and the negations are only applied to **variables**.

Example 1.4.4

The formula $\lnot a \lor b$ is in normal form

$a \Rightarrow b$ is not in normal form, even if it is equivalent to the first one.
Normal form

**Definition 1.4.3**

A formula is in normal form if it only contains the operators $\land$, $\lor$, $\neg$ and the negations are only applied to variables.

**Example 1.4.4**

The formula $\neg a \lor b$ is in normal form.

$a \Rightarrow b$ is not in normal form, even if it is equivalent to the first one.

Every formula admits an equivalent normal form.
Computing a normal form

1. **Equivalence elimination**

   - Replace any occurrence of $A \iff B$ by
     - (a) $(\neg A \lor B) \land (\neg B \lor A)$
     - OR
     - (b) $(A \land B) \lor (\neg A \land \neg B)$

2. **Implication elimination**

3. **Shifting negations towards variables**

   - (a) $\neg \neg A$ by $A$
   - (b) $\neg (A \lor B)$ by $\neg A \land \neg B$
   - (c) $\neg (A \land B)$ by $\neg A \lor \neg B$
Computing a normal form

1. **Equivalence elimination**
   Replace any occurrence of $A \iff B$ by
   
   (a) $(\neg A \lor B) \land (\neg B \lor A)$
   
   OR
   
   (b) $(A \land B) \lor (\neg A \land \neg B)$

2. **Implication elimination**

3. **Shifting negations towards variables**
Computing a normal form

1. **Equivalence elimination**
   Replace any occurrence of $A \iff B$ by
   
   (a) $(\neg A \lor B) \land (\neg B \lor A)$
   OR
   
   (b) $(A \land B) \lor (\neg A \land \neg B)$

2. **Implication elimination**
   Replace any occurrence of $A \Rightarrow B$ by $\neg A \lor B$

3. **Shifting negations towards variables**
Computing a normal form

1. **Equivalence elimination**
   Replace any occurrence of $A \iff B$ by
   - (a) $(\neg A \lor B) \land (\neg B \lor A)$
   - OR
   - (b) $(A \land B) \lor (\neg A \land \neg B)$

2. **Implication elimination**
   Replace any occurrence of $A \Rightarrow B$ by $\neg A \lor B$

3. **Shifting negations towards variables**
   Replace any occurrence of
   - (a) $\neg \neg A$ by $A$
   - (b) $\neg (A \lor B)$ by $\neg A \land \neg B$
   - (c) $\neg (A \land B)$ by $\neg A \lor \neg B$
Remark 1.4.5: simplifications

Simplify as soon as possible:

1. Replace \( \neg (A \Rightarrow B) \) by \( A \land \neg B \).
Remark 1.4.5: simplifications

Simplify as soon as possible:

1. Replace \( \neg (A \Rightarrow B) \) by \( A \land \neg B \).
2. Replacing a conjunction by 0 if it contains a formula and its negation
3. Replace a disjunction by 1 if it contains a formula and its negation
Remark 1.4.5: simplifications

Simplify as soon as possible:

1. Replace $\neg(A \Rightarrow B)$ by $A \land \neg B$.

2. Replacing a conjunction by 0 if it contains a formula and its negation

3. Replace a disjunction by 1 if it contains a formula and its negation

4. Apply:
   - Idempotence of $\land$ and $\lor$
   - Neutrality and absorption of 0 and 1
   - Replace $\neg 1$ by 0 and vice versa.
Remark 1.4.5: simplifications

Simplify as soon as possible:

1. Replace $\neg (A \Rightarrow B)$ by $A \land \neg B$.
2. Replacing a conjunction by 0 if it contains a formula and its negation
3. Replace a disjunction by 1 if it contains a formula and its negation
4. Apply :
   - Idempotence of $\land$ and $\lor$
   - Neutrality and absorption of 0 and 1
   - Replace $\neg 1$ by 0 and vice versa.
5. Apply the simplifications:
   - $x \lor (x \land y) \equiv x$,
   - $x \land (x \lor y) \equiv x$,
   - $x \lor (\neg x \land y) \equiv x \lor y$
Disjunctive normal form (DNF)

**Definition 1.4.6**

A formula is in disjunctive normal form (DNF) if and only if it is a disjunction (sum) of monomials.

Method: distribute the conjunctions over the disjunctions

\[ x \land (y \lor z) \equiv (x \land y) \lor (x \land z) \]
Disjunctive normal form (DNF)

Definition 1.4.6

A formula is in disjunctive normal form (DNF) if and only if it is a disjunction (sum) of monomials.

Method: distribute the conjunctions over the disjunctions

\[ x \land (y \lor z) \equiv (x \land y) \lor (x \land z) \]

The interest of a DNF is to highlight the models.
Disjunctive normal form (DNF)

Definition 1.4.6

A formula is in disjunctive normal form (DNF) if and only if it is a disjunction (sum) of monomials.

Method: distribute the conjunctions over the disjunctions

\[ x \land (y \lor z) \equiv (x \land y) \lor (x \land z) \]

The interest of a DNF is to highlight the models.

Example 1.4.7

\((x \land y) \lor (\neg x \land \neg y \land z)\) is a DNF, which has two main models:
Disjunctive normal form (DNF)

**Definition 1.4.6**

A formula is in *disjunctive normal form (DNF)* if and only if it is a disjunction (sum) of monomials.

Method: distribute the conjunctions over the disjunctions

\[ x \land (y \lor z) \equiv (x \land y) \lor (x \land z) \]

The interest of a DNF is to highlight the models.

**Example 1.4.7**

\( (x \land y) \lor (\neg x \land \neg y \land z) \) is a DNF, which has two main models:

- \( x \mapsto 1, y \mapsto 1 \)
- \( x \mapsto 0, y \mapsto 0, z \mapsto 1 \)
Conjunctive normal form (CNF)

Definition 1.4.11

A formula is a conjunctive normal form (CNF) if and only if it is a conjunction (product) of clauses.

Apply the (unusual) distributivity of disjunction over conjunction:

\[ A \lor (B \land C) \equiv (A \lor B) \land (A \lor C) \]
Conjunctive normal form (CNF)

Definition 1.4.11

A formula is a conjunctive normal form (CNF) if and only if it is a conjunction (product) of clauses.

Apply the (unusual) distributivity of disjunction over conjunction:

\[ A \lor (B \land C) \equiv (A \lor B) \land (A \lor C) \]

The interest of a CNF is to highlight the counter-models.
Conjunctive normal form (CNF)

Definition 1.4.11

A formula is a conjunctive normal form (CNF) if and only if it is a conjunction (product) of clauses.

Apply the (unusual) distributivity of disjunction over conjunction:

\[ A \lor (B \land C) \equiv (A \lor B) \land (A \lor C) \]

The interest of a CNF is to highlight the counter-models.

Example 1.4.12

\((x \lor y) \land (\neg x \lor \neg y \lor z)\) is a CNF, which has two counter-models.
Conjunctive normal form (CNF)

Definition 1.4.11

A formula is a conjunctive normal form (CNF) if and only if it is a conjunction (product) of clauses.

Apply the (unusual) distributivity of disjunction over conjunction:

\[ A \lor (B \land C) \equiv (A \lor B) \land (A \lor C) \]

The interest of a CNF is to highlight the counter-models.

Example 1.4.12

\[(x \lor y) \land (\neg x \lor \neg y \lor z)\]
is a CNF, which has two counter-models.

\[
\begin{align*}
&x \mapsto 0, y \mapsto 0 \\
&x \mapsto 1, y \mapsto 1, z \mapsto 0.
\end{align*}
\]
Examples 1.4.8 and 1.4.13

Transformation in **DNF** of the following:

\[ (a \lor b) \land (c \lor d \lor e) \equiv \]
Examples 1.4.8 and 1.4.13

Transformation in DNF of the following:

\[(a \lor b) \land (c \lor d \lor e) \equiv \]

\[(a \land c) \lor (a \land d) \lor (a \land e) \lor (b \land c) \lor (b \land d) \lor (b \land e).\]
Examples 1.4.8 and 1.4.13

Transformation in DNF of the following:

\[(a \lor b) \land (c \lor d \lor e) \equiv\]

\[(a \land c) \lor (a \land d) \lor (a \land e) \lor (b \land c) \lor (b \land d) \lor (b \land e).\]

Transformation in CNF of the following:

\[(a \land b) \lor (c \land d \land e) \equiv\]
Examples 1.4.8 and 1.4.13

Transformation in DNF of the following:

\[(a \lor b) \land (c \lor d \lor e) \equiv (a \land c) \lor (a \land d) \lor (a \land e) \lor (b \land c) \lor (b \land d) \lor (b \land e).\]

Transformation in CNF of the following:

\[(a \land b) \lor (c \land d \land e) \equiv (a \lor c) \land (a \lor d) \land (a \lor e) \land (b \lor c) \land (b \lor d) \land (b \lor e).\]
Another use of DNFs

Transforming a formula into a disjunction of monomials also allows us to determine whether the formula is valid or not.
Another use of DNFs

Transforming a formula into a disjunction of monomials also allows us to determine whether the formula is valid or not.

Let $A$ be a formula whose validity we wish to check:
Another use of DNFs

Transforming a formula into a disjunction of monomials also allows us to determine whether the formula is valid or not.

Let \( A \) be a formula whose validity we wish to check:

We transform \( \neg A \) in an equivalent disjunction of monomials \( B \):
Another use of DNFs

Transforming a formula into a disjunction of monomials also allows us to determine whether the formula is valid or not.

Let $A$ be a formula whose validity we wish to check:

We transform $\neg A$ in an equivalent disjunction of monomials $B$:

- If $B = 0$ then $\neg A = 0$, hence $A = 1$, that is, $A$ is valid
Another use of DNFs

Transforming a formula into a disjunction of monomials also allows us to determine whether the formula is valid or not.

Let $A$ be a formula whose validity we wish to check:

We transform $\neg A$ in an equivalent disjunction of monomials $B$:

- If $B = 0$ then $\neg A = 0$, hence $A = 1$, that is, $A$ is valid
- Otherwise $B$ is equal to a disjunction of nonzero monomials equivalent to $\neg A$, which give us models of $\neg A$, hence counter-models of $A$. 
Example 1.4.9

Let $A = (p \Rightarrow (q \Rightarrow r)) \Rightarrow (p \land q \Rightarrow r)$

Determine whether $A$ is valid.

$\neg A$
Example 1.4.9

Let $A = (p \Rightarrow (q \Rightarrow r)) \Rightarrow (p \land q \Rightarrow r)$

Determine whether $A$ is valid.

\[
\neg A = (p \Rightarrow (q \Rightarrow r)) \land \neg (p \land q \Rightarrow r)
\]

since $\neg (B \Rightarrow C) \equiv B \land \neg C$
Example 1.4.9

Let \( A = (p \Rightarrow (q \Rightarrow r)) \Rightarrow (p \land q \Rightarrow r) \)

Determine whether \( A \) is valid.

\[
\neg A \\
\equiv (p \Rightarrow (q \Rightarrow r)) \land \neg (p \land q \Rightarrow r) \\
\equiv (\neg p \lor \neg q \lor r) \land \neg (p \land q \Rightarrow r)
\]

since \( \neg (B \Rightarrow C) \equiv B \land \neg C \)

eliminating two \( \Rightarrow \)

Hence \( \neg A = 0 \) and \( A = 1 \), that is \( A \) is valid.
Example 1.4.9

Let \( A = (p \Rightarrow (q \Rightarrow r)) \Rightarrow (p \land q \Rightarrow r) \)

Determine whether \( A \) is valid.

\[
\neg A \\
\equiv (p \Rightarrow (q \Rightarrow r)) \land \neg (p \land q \Rightarrow r) \quad \text{ since } \neg (B \Rightarrow C) \equiv B \land \neg C \\
\equiv (\neg p \lor \neg q \lor r) \land \neg (p \land q \Rightarrow r) \quad \text{ eliminating two } \Rightarrow \\
\equiv (\neg p \lor \neg q \lor r) \land (p \land q \land \neg r) \quad \text{ since } \neg (B \Rightarrow C) \equiv B \land \neg C 
\]
Example 1.4.9

Let $A = (p \Rightarrow (q \Rightarrow r)) \Rightarrow (p \land q \Rightarrow r)$

Determine whether $A$ is valid.

\[
\neg A \\
\equiv (p \Rightarrow (q \Rightarrow r)) \land \neg(p \land q \Rightarrow r) \\
\equiv (\neg p \lor \neg q \lor r) \land \neg(p \land q \Rightarrow r) \quad \text{since } \neg(B \Rightarrow C) \equiv B \land \neg C \\
\equiv (\neg p \lor \neg q \lor r) \land (p \land q \land \neg r) \quad \text{eliminating two } \Rightarrow \\
\equiv (\neg q \lor r) \land p \land q \land \neg r \quad \text{since } \neg(B \Rightarrow C) \equiv B \land \neg C \\
\equiv (\neg q \lor r) \land p \land q \land \neg r \quad \text{simplification } x \land (\neg x \lor y)
\]
Example 1.4.9

Let \( A = (p \Rightarrow (q \Rightarrow r)) \Rightarrow (p \land q \Rightarrow r) \)

Determine whether \( A \) is valid.

\[
\neg A \\
\equiv (p \Rightarrow (q \Rightarrow r)) \land \neg (p \land q \Rightarrow r) \\
\equiv (\neg p \lor \neg q \lor r) \land \neg (p \land q \Rightarrow r) \\
\equiv (\neg p \lor \neg q \lor r) \land (p \land q \land \neg r) \\
\equiv (\neg q \lor r) \land p \land q \land \neg r \\
\equiv (r) \land p \land q \land \neg r
\]

since \( \neg (B \Rightarrow C) \equiv B \land \neg C \)

eliminating two \( \Rightarrow \)

simplification \( x \land (\neg x \lor y) \)

simplification \( x \land (\neg x \lor y) \)

Hence \( \neg A = 0 \) and \( A = 1 \), that is \( A \) is valid.
Example 1.4.9

Let $A = (p \Rightarrow (q \Rightarrow r)) \Rightarrow (p \land q \Rightarrow r)$

Determine whether $A$ is valid.

\[
\neg A = (p \Rightarrow (q \Rightarrow r)) \land \neg (p \land q \Rightarrow r)
\]

\[
\equiv (\neg p \lor \neg q \lor r) \land \neg (p \land q \Rightarrow r) \quad \text{since } \neg (B \Rightarrow C) \equiv B \land \neg C
\]

\[
\equiv (\neg p \lor \neg q \lor r) \land (p \land q \land \neg r) \quad \text{eliminating two } \Rightarrow
\]

\[
\equiv (\neg q \lor r) \land p \land q \land \neg r \quad \text{since } \neg (B \Rightarrow C) \equiv B \land \neg C
\]

\[
\equiv (r) \land p \land q \land \neg r \quad \text{simplification } x \land (\neg x \lor y)
\]

\[
= 0
\]

Hence $\neg A = 0$ and $A = 1$, that is $A$ is valid.
Example 1.4.10

Let \( A = (a \Rightarrow b) \land c \lor (a \land d) \).

Determine whether \( A \) is valid.

\[ \neg A \]
Example 1.4.10

Let $A = (a \Rightarrow b) \land c \lor (a \land d)$.

Determine whether $A$ is valid.

\[ \neg A \]
\[ \equiv \neg((a \Rightarrow b) \land c) \land \neg(a \land d) \quad \text{(de Morgan)} \]

We obtain 3 models of $\neg A$: $(a \mapsto \top, b \mapsto \bot, d \mapsto \bot)$, $(a \mapsto \bot, c \mapsto \bot)$, $(c \mapsto \bot, d \mapsto \bot)$. That is, counter-models of $A$.

Hence $A$ is not valid.
Example 1.4.10

Let $A = (a \Rightarrow b) \land c \lor (a \land d)$.

Determine whether $A$ is valid.

$\neg A$

$\equiv \neg((a \Rightarrow b) \land c) \land \neg(a \land d)$ (de Morgan)

$\equiv (\neg(a \Rightarrow b) \lor \neg c) \land (\neg a \lor \neg d)$ (de Morgan)
Example 1.4.10

Let $A = (a \Rightarrow b) \land c \lor (a \land d)$.

Determine whether $A$ is valid.

$\neg A$

$\equiv \neg ((a \Rightarrow b) \land c) \land \neg (a \land d)$ (de Morgan)

$\equiv (\neg (a \Rightarrow b) \lor \neg c) \land (\neg a \lor \neg d)$ (de Morgan)

$\equiv ((a \land \neg b) \lor \neg c) \land (\neg a \lor \neg d)$ elimination of the implication
Example 1.4.10

Let \( A = (a \Rightarrow b) \land c \lor (a \land d) \).

Determine whether \( A \) is valid.

\[
\neg A \\
\equiv \neg((a \Rightarrow b) \land c) \land \neg(a \land d) \quad \text{(de Morgan)} \\
\equiv (\neg(a \Rightarrow b) \lor \neg c) \land (\neg a \lor \neg d) \quad \text{(de Morgan)} \\
\equiv ((a \land \neg b) \lor \neg c) \land (\neg a \lor \neg d) \quad \text{elimination of the implication} \\
\equiv (a \land \neg b \land \neg a) \lor (a \land \neg b \land \neg d) \lor (\neg c \land \neg a) \lor (\neg c \land \neg d) \quad \text{distributivity of } \lor \text{ over } \land
\]
Example 1.4.10

Let $A = (a \Rightarrow b) \land c \lor (a \land d)$.

Determine whether $A$ is valid.

\[
\neg A \\
\equiv \neg((a \Rightarrow b) \land c) \land \neg(a \land d) \\
\equiv (\neg(a \Rightarrow b) \lor \neg c) \land (\neg a \lor \neg d) \\
\equiv ((a \land \neg b) \lor \neg c) \land (\neg a \lor \neg d) \\
\equiv (a \land \neg b \land \neg a) \lor (a \land \neg b \land \neg d) \lor (\neg c \land \neg a) \lor (\neg c \land \neg d) \\
\equiv (a \land \neg b \land \neg d) \lor (\neg c \land \neg a) \lor (\neg c \land \neg d) \\
\]

We obtain 3 models of $\neg A$: $(a \mapsto 1, b \mapsto 0, d \mapsto 0)$, $(a \mapsto 0, c \mapsto 0)$, $(c \mapsto 0, d \mapsto 0)$. That is, counter-models of $A$. Hence $A$ is not valid.
Example 1.4.10

Let $A = (a \Rightarrow b) \land c \lor (a \land d)$.

Determine whether $A$ is valid.

$\neg A$

$\equiv \neg((a \Rightarrow b) \land c) \land \neg(a \land d)$ (de Morgan)

$\equiv ((\neg a \lor \neg b) \lor \neg c) \land (\neg a \lor \neg d)$ (de Morgan)

$\equiv ((a \land \neg b) \lor \neg c) \land (\neg a \lor \neg d)$ elimination of the implication

$\equiv (a \land \neg b \land \neg a) \lor (a \land \neg b \land \neg d) \lor (\neg c \land \neg a) \lor (\neg c \land \neg d)$ distributivity of $\lor$ over $\land$

$\equiv (a \land \neg b \land \neg d) \lor (\neg c \land \neg a) \lor (\neg c \land \neg d)$ 1st monomial contradictory

We obtain 3 models of $\neg A$: $(a \mapsto 1, b \mapsto 0, d \mapsto 0)$, $(a \mapsto 0, c \mapsto 0)$,
$(c \mapsto 0, d \mapsto 0)$.
That is, counter-models of $A$.
Hence $A$ is not valid.
Plan

Consequence

Important equivalences

Substitution and replacement

Normal forms

Conclusion
Today

- **Substitutions** allow us to deduce the validity of a formula from another.

- **Replacements** allow us to change part of a formula without changing its meaning and thus allow us to compute a simpler equivalent formula.

- Every formula admits normal forms which allow to highlight its models and counter-models.
Next course

- Boolean algebra
- Boolean functions
- Resolution

Prove our example by formula simplification

\[(p \Rightarrow \neg j) \land (\neg p \Rightarrow j) \land (j \Rightarrow m) \Rightarrow m \lor p\]