Chapitre 2
Définition de la logique propositionnelle
Cours 2
Syntaxe de la logique propositionnelle et preuves par induction

Contenu de ce chapitre

- Syntaxe de la logique propositionnelle
  - Définitions inductives
  - Preuves par induction
- Sémantique de la logique propositionnelle
  - Définition de fonctions par récurrence
- Algorithmique de la logique propositionnelle (première approche)

Contenu de ce chapitre

- Syntaxe de la logique propositionnelle
  - Définitions inductives
  - Preuves par induction
- Sémantique de la logique propositionnelle
  - Définition de fonctions par récurrence
- Algorithmique de la logique propositionnelle (première approche)

Contenu de ce chapitre

- Syntaxe de la logique propositionnelle
  - Définitions inductives
  - Preuves par induction
- Sémantique de la logique propositionnelle
  - Définition de fonctions par récurrence
- Algorithmique de la logique propositionnelle (première approche)

Vers une définition de la syntaxe

Première Idée

On définit les formules selon les différents cas de leur forme.

Une formule propositionnelle

- peut être une variable propositionnelle
- $\neg p$, où p est une formule
- $(p \wedge q)$, où p et q sont des formules
- $(p \vee q)$, où p et q sont des formules

Une précision importante :

Seulement les chaînes de caractères ainsi formées sont des formules.

Vers une définition de la syntaxe

Première Idée

On définit les formules selon les différents cas de leur forme.

Une formule propositionnelle

- peut être une variable propositionnelle
- $\neg p$, où p est une formule
- $(p \wedge q)$, où p et q sont des formules
- $(p \vee q)$, où p et q sont des formules

Une précision importante :

Seulement les chaînes de caractères ainsi formées sont des formules.

Vers une définition de la syntaxe

Première Idée

On définit les formules selon les différents cas de leur forme.

Une formule propositionnelle

- peut être une variable propositionnelle
- $\neg p$, où p est une formule
- $(p \wedge q)$, où p et q sont des formules
- $(p \vee q)$, où p et q sont des formules

Une précision importante :

Seulement les chaînes de caractères ainsi formées sont des formules.

Exemples de formules propositionnelles

Sont des formules :

- x (car variable propositionnelle)
- $\neg x$ (car négation d'une formule)
- y (car variable propositionnelle)
- $(\neg x \wedge y)$ (car conjonction de deux formules)

Ne sont pas des formules (dans le sens stricte) :

- $((x$
- $x \vee y$

Exemples de formules propositionnelles

Sont des formules :

- x (car variable propositionnelle)
- $\neg x$ (car négation d'une formule)
- y (car variable propositionnelle)
- $(\neg x \wedge y)$ (car conjonction de deux formules)

Ne sont pas des formules (dans le sens stricte) :

- $((x$
- $x \vee y$

Exemples de formules propositionnelles

Sont des formules :

- x (car variable propositionnelle)
- $\neg x$ (car négation d'une formule)
- y (car variable propositionnelle)
- $(\neg x \wedge y)$ (car conjonction de deux formules)

Ne sont pas des formules (dans le sens stricte) :

- $((x$
- $x \vee y$

Exemples de formules propositionnelles

Sont des formules :

- x (car variable propositionnelle)
- $\neg x$ (car négation d'une formule)
- y (car variable propositionnelle)
- $(\neg x \wedge y)$ (car conjonction de deux formules)

Ne sont pas des formules (dans le sens stricte) :

- $((x$
- $x \vee y$

Exemples de formules propositionnelles

Sont des formules :

- x (car variable propositionnelle)
- $\neg x$ (car négation d'une formule)
- y (car variable propositionnelle)
- $(\neg x \wedge y)$ (car conjonction de deux formules)

Ne sont pas des formules (dans le sens stricte) :

- $((x$
- $x \vee y$

Exemples de formules propositionnelles

Sont des formules :

- x (car variable propositionnelle)
- $\neg x$ (car négation d'une formule)
- y (car variable propositionnelle)
- $(\neg x \wedge y)$ (car conjonction de deux formules)

Ne sont pas des formules (dans le sens stricte) :

- $((x$
- $x \vee y$

Exemples de formules propositionnelles

Sont des formules :

- x (car variable propositionnelle)
- $\neg x$ (car négation d'une formule)
- y (car variable propositionnelle)
- $(\neg x \wedge y)$ (car conjonction de deux formules)

Ne sont pas des formules (dans le sens stricte) :

- $((x$
- $x \vee y$

Les variables propositionnelles

On choisit d'abord un ensemble de variables propositionnelles :

$$V := \{x, x_1, x_2, x_3, \ldots, y, y_1, y_2, \ldots, z, z_1, z_2, z_3, \ldots\}$$

Nous admettons également les décorations habituelles des variables propositionnelles comme par exemple x', y''.

Tentative de définition

L'ensemble *Form* des formules propositionnelles est défini comme l'ensemble de chaînes de caractères tel que :

1. $V \subseteq Form$
2. Si $p \in Form$ alors $\neg p \in Form$
3. Si $p, q \in Form$ alors $(p \wedge q) \in Form$
4. Si $p, q \in Form$ alors $(p \vee q) \in Form$

Nous appelons ces quatre conditions (1) - (4) les propriétés de clôture des formules propositionnelles.

Le problème de la tentative de définition

Il y a plusieurs ensembles *Form* qui satisfont la description :

- L'ensemble E_1 des formules propositionnelles dans le sens de la description informelle au-dessus.
- L'ensemble E_2 de toutes les chaînes de caractères qui ont le même nombre de « (» que de «) ».
  Mais cet ensemble contient aussi la chaîne « ()() ».
- L'ensemble E_3 de toutes les chaînes de caractères.
  Cet ensemble contient aussi la chaîne « ((».

Nous avons $E_1 \subseteq E_2 \subseteq E_3$.

Le problème de la tentative de définition

Il y a plusieurs ensembles *Form* qui satisfont la description :

- L'ensemble E_1 des formules propositionnelles dans le sens de la description informelle au-dessus.
- L'ensemble E_2 de toutes les chaînes de caractères qui ont le même nombre de « (» que de «) ».
  Mais cet ensemble contient aussi la chaîne « ()() ».
- L'ensemble E_3 de toutes les chaînes de caractères.
  Cet ensemble contient aussi la chaîne « ((».

Nous avons $E_1 \subseteq E_2 \subseteq E_3$.

Le problème de la tentative de définition

Il y a plusieurs ensembles *Form* qui satisfont la description :

- L'ensemble E_1 des formules propositionnelles dans le sens de la description informelle au-dessus.
- L'ensemble E_2 de toutes les chaînes de caractères qui ont le même nombre de « (» que de «) ».
  Mais cet ensemble contient aussi la chaîne « ()() ».
- L'ensemble E_3 de toutes les chaînes de caractères.
  Cet ensemble contient aussi la chaîne « ((».

Nous avons $E_1 \subseteq E_2 \subseteq E_3$.

Le problème de la tentative de définition

Il y a plusieurs ensembles *Form* qui satisfont la description :

- L'ensemble E_1 des formules propositionnelles dans le sens de la description informelle au-dessus.
- L'ensemble E_2 de toutes les chaînes de caractères qui ont le même nombre de « (» que de «) ».
  Mais cet ensemble contient aussi la chaîne « ()() ».
- L'ensemble E_3 de toutes les chaînes de caractères.
  Cet ensemble contient aussi la chaîne « ((».

Nous avons $E_1 \subseteq E_2 \subseteq E_3$.

La racine du problème

Notre tentative de définition prend en compte

- le cas de base (les variables propositionnelles)
- et les constructions autorisées ($\neg$, $\wedge$, $\vee$)

mais ne prend absolument pas en compte la restriction qu'une chaîne qui ne peut pas être construite selon les règles précédentes n'est pas une formule.

La bonne définition

Definition

L'ensemble *Form* des formules propositionnelles est le plus petit ensemble de chaînes de caractères tel que :

1. $V \subseteq Form$
2. Si $p \in Form$ alors $\neg p \in Form$
3. Si $p, q \in Form$ alors $(p \wedge q) \in Form$
4. Si $p, q \in Form$ alors $(p \vee q) \in Form$

Définitions inductives

Le principe

- On a certaines propriétés de clôture (ici : les quatre propriétés de clôture des formules propositionnelles)
- On définit un ensemble (ici : *Form*) comme étant le plus petit ensemble ayant ces propriétés de clôture.

Qu'est-ce que c'est le plus petit ensemble ?

Si un ensemble X satisfait toutes les propriétés de clôture des formules propositionnelles, alors $\mathit{Form} \subseteq X$.

As-t-on le droit d'écrire une telle définition ?

1. Est-ce qu'il existe effectivement (au moins) un tel plus petit ensemble ?
2. Cet ensemble, est-il unique ?

Définitions inductives

Le principe

- On a certaines propriétés de clôture (ici : les quatre propriétés de clôture des formules propositionnelles)
- On définit un ensemble (ici : *Form*) comme étant le plus petit ensemble ayant ces propriétés de clôture.

Qu'est-ce que c'est le plus petit ensemble ?

Si un ensemble X satisfait toutes les propriétés de clôture des formules propositionnelles, alors $Form \subseteq X$.

As-t-on le droit d'écrire une telle définition ?

1. Est-ce qu'il existe effectivement (au moins) un tel plus petit ensemble ?
2. Cet ensemble, est-il unique ?

Définitions inductives

Le principe

- On a certaines propriétés de clôture (ici : les quatre propriétés de clôture des formules propositionnelles)
- On définit un ensemble (ici : *Form*) comme étant le plus petit ensemble ayant ces propriétés de clôture.

Qu'est-ce que c'est le plus petit ensemble ?

Si un ensemble X satisfait toutes les propriétés de clôture des formules propositionnelles, alors $Form \subseteq X$.

As-t-on le droit d'écrire une telle définition ?

1. Est-ce qu'il existe effectivement (au moins) un tel plus petit ensemble ?
2. Cet ensemble, est-il unique ?

Existence d'un plus petit ensemble

- A priori, il n'est pas évident qu'il existe un plus petit ensemble avec une certaine propriété donnée.
- Exemple : le plus petit ensemble d'entiers qui contient au moins deux éléments ?
- Ensembles $\{0, 1\}$ et $\{2, 3\}$
- Dans notre cas, il y a un plus petit ensemble (mais on ne va pas le montrer).
- En bref : c'est dû au fait qu'il s'agit ici de propriétés de clôture.

Existence d'un plus petit ensemble

- A priori, il n'est pas évident qu'il existe un plus petit ensemble avec une certaine propriété donnée.
- Exemple : le plus petit ensemble d'entiers qui contient au moins deux éléments ?
- Ensembles $\{0, 1\}$ et $\{2, 3\}$
- Dans notre cas, il y a un plus petit ensemble (mais on ne va pas le montrer).
- En bref : c'est dû au fait qu'il s'agit ici de propriétés de clôture.

Existence d'un plus petit ensemble

- A priori, il n'est pas évident qu'il existe un plus petit ensemble avec une certaine propriété donnée.
- Exemple : le plus petit ensemble d'entiers qui contient au moins deux éléments ?
- Ensembles $\{0, 1\}$ et $\{2, 3\}$
- Dans notre cas, il y a un plus petit ensemble (mais on ne va pas le montrer).
- En bref : c'est dû au fait qu'il s'agit ici de propriétés de clôture.

Existence d'un plus petit ensemble

- A priori, il n'est pas évident qu'il existe un plus petit ensemble avec une certaine propriété donnée.
- Exemple : le plus petit ensemble d'entiers qui contient au moins deux éléments ?
- Ensembles $\{0, 1\}$ et $\{2, 3\}$
- Dans notre cas, il y a un plus petit ensemble (mais on ne va pas le montrer).
- En bref : c'est dû au fait qu'il s'agit ici de propriétés de clôture.

Existence d'un plus petit ensemble

- A priori, il n'est pas évident qu'il existe un plus petit ensemble avec une certaine propriété donnée.
- Exemple : le plus petit ensemble d'entiers qui contient au moins deux éléments ?
- Ensembles $\{0, 1\}$ et $\{2, 3\}$
- Dans notre cas, il y a un plus petit ensemble (mais on ne va pas le montrer).
- En bref : c'est dû au fait qu'il s'agit ici de propriétés de clôture.

Unicité du plus petit ensemble

- Supposons que E_1 et E_2 sont des plus petits ensembles avec une propriété donnée.
- Donc, $E_1 \subseteq E_2$
- et aussi $E_2 \subseteq E_1$
- Conséquence : $E_1 = E_2$

Unicité du plus petit ensemble

- Supposons que E_1 et E_2 sont des plus petits ensembles avec une propriété donnée.
- Donc, $E_1 \subseteq E_2$
- et aussi $E_2 \subseteq E_1$
- Conséquence : $E_1 = E_2$

Unicité du plus petit ensemble

- Supposons que E_1 et E_2 sont des plus petits ensembles avec une propriété donnée.
- Donc, $E_1 \subseteq E_2$
- et aussi $E_2 \subseteq E_1$
- Conséquence : $E_1 = E_2$

Unicité du plus petit ensemble

- Supposons que E_1 et E_2 sont des plus petits ensembles avec une propriété donnée.
- Donc, $E_1 \subseteq E_2$
- et aussi $E_2 \subseteq E_1$
- Conséquence : $E_1 = E_2$

Définitions inductive

- Notre définition de la syntaxe de la logique propositionnelle est une définition inductive.
- On verra d'autres exemples de définitions inductives.
- Ici on ne s'intéresse pas à étudier les définition inductives en général, il suffit de dire qu'elles sont toujours de la forme «le plus petit ensemble qui satisfait certaines propriétés de clôture».

Définitions inductive

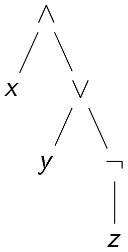
- Notre définition de la syntaxe de la logique propositionnelle est une définition inductive.
- On verra d'autres exemples de définitions inductives.
- Ici on ne s'intéresse pas à étudier les définition inductives en général, il suffit de dire qu'elles sont toujours de la forme «le plus petit ensemble qui satisfait certaines propriétés de clôture».

Définitions inductive

- Notre définition de la syntaxe de la logique propositionnelle est une définition inductive.
- On verra d'autres exemples de définitions inductives.
- Ici on ne s'intéresse pas à étudier les définition inductives en général, il suffit de dire qu'elles sont toujours de la forme «le plus petit ensemble qui satisfait certaines propriétés de clôture».

Syntaxe Abstraite

Syntaxe abstraite de $(x \wedge (y \vee \neg z))$:



Structure de la formule.

Encodage de l'implication

On peut encoder l'implication de la manière suivante:

$$(p \Rightarrow q) := (\neg p \vee q)$$
$$(p \Leftrightarrow q) := ((p \Rightarrow q) \wedge (q \Rightarrow p))$$

On l'utilisera comme suit:

si p alors q	$(p \Rightarrow q)$
q si p	$(p \Rightarrow q)$
q seulement si p	$(p \Leftarrow q)$ ou $(q \Rightarrow p)$
seulement si p q	$(p \Leftarrow q)$ ou $(q \Rightarrow p)$
q uniquement si p	$(p \Leftarrow q)$ ou $(q \Rightarrow p)$
uniquement si p q	$(p \Leftarrow q)$ ou $(q \Rightarrow p)$
q si et seulement si p	$(p \Leftrightarrow q)$

Preuves par induction : Un exemple

Théorème : Toute formule propositionnelle a le même nombre de parenthèses ouvrantes que de parenthèses fermantes.

Notation

- $|w|_($ dénote le nombre de parenthèses ouvrantes de w.
- $|w|_)$ dénote le nombre de parenthèses fermantes de w.
- Par exemple, $|(x \wedge y)|_(= 1$
- Pareil pour des autres symboles : $|w|_\neg$, etc.

À montrer : pour tout $w \in Form$, $|w|_(= |w|_)$.

Notation

- $|w|_($ dénote le nombre de parenthèses ouvrantes de w.
- $|w|_)$ dénote le nombre de parenthèses fermantes de w.
- Par exemple, $|(x \wedge y)|_(= 1$
- Pareil pour des autres symboles : $|w|_\neg$, etc.

À montrer : pour tout $w \in \mathit{Form}$, $|w|_(= |w|_)$.

Notation

- $|w|_($ dénote le nombre de parenthèses ouvrantes de w.
- $|w|_)$ dénote le nombre de parenthèses fermantes de w.
- Par exemple, $|(x \wedge y)|_(= 1$
- Pareil pour des autres symboles : $|w|_\neg$, etc.

À montrer : pour tout $w \in Form$, $|w|_(= |w|_)$.

Notation

- $|w|_($ dénote le nombre de parenthèses ouvrantes de w.
- $|w|_)$ dénote le nombre de parenthèses fermantes de w.
- Par exemple, $|(x \wedge y)|_(= 1$
- Pareil pour des autres symboles : $|w|_\neg$, etc.

À montrer : pour tout $w \in \mathit{Form}$, $|w|_(= |w|_)$.

Notation

- $|w|_($ dénote le nombre de parenthèses ouvrantes de w.
- $|w|_)$ dénote le nombre de parenthèses fermantes de w.
- Par exemple, $|(x \wedge y)|_(= 1$
- Pareil pour des autres symboles : $|w|_\neg$, etc.

À montrer : pour tout $w \in \mathit{Form}$, $|w|_(= |w|_)$.

Le principe d'une preuve par induction structurelle

Le principe est illustré ici sur l'exemple des formules propositionnelles :

Théorème : Soit P une propriété sur les chaînes de caractères. Supposons que P vérifie les énoncés de clôture suivants :

- tout élément de V a la propriété P,
- si p satisfait P alors $\neg p$ satisfait P,
- si p et q satisfont P alors $(p \wedge q)$ satisfait P,
- si p et q satisfont P alors $(p \vee q)$ satisfait P,

Alors tous les éléments de *Form* satisfont P.

Exemple de propriété P : avoir le même nombre de (que de).

Comment rédiger une preuve par induction structurelle ?

À montrer :

pour tout $w \in Form$, $|w|_(= |w|_)$.

(1) Cas des variables :

À montrer : $|w|_(= |w|_)$ pour tout $w \in V$.

Démonstration :

Si $w \in V$ alors $|w|_(= 0$ et $|w|_) = 0$, donc $|w|_(= |w|_)$.

Comment rédiger une preuve par induction structurelle ?

À montrer :

pour tout $w \in Form$, $|w|_(= |w|_)$.

(1) Cas des variables :

À montrer : $|w|_(= |w|_)$ pour tout $w \in V$.

Démonstration :

Si $w \in V$ alors $|w|_(= 0$ et $|w|_) = 0$, donc $|w|_(= |w|_)$.

Comment rédiger une preuve par induction structurelle ?

À montrer :

pour tout $w \in Form$, $|w|_(= |w|_)$.

(1) Cas des variables :

À montrer : $|w|_(= |w|_)$ pour tout $w \in V$.

Démonstration :

Si $w \in V$ alors $|w|_(= 0$ et $|w|_) = 0$, donc $|w|_(= |w|_)$.

Rédaction d'une preuve par induction (2)

(2) Cas de la négation :

Soit $p \in Form$. Hypothèse d'induction : $|p|_(= |p|_)$.
À montrer : $|\neg p|_(= |\neg p|_)$.

Démonstration :

$$\begin{array}{rcll} |\neg p|_(& = & |p|_(& \text{par définition } |\cdot|_(\\ & = & |p|_) & \text{par hypothèse d'induction} \\ & = & |\neg p|_) & \text{par définition } |\cdot|_) \end{array}$$

Rédaction d'une preuve par induction (2)

(2) Cas de la négation :

Soit $p \in Form$. Hypothèse d'induction : $|p|_(= |p|_)$.
À montrer : $|\neg p|_(= |\neg p|_)$.

Démonstration :

$$
\begin{aligned}
|\neg p|_(&= |p|_(&& \text{par définition } |\cdot|_(\\
&= |p|_) && \text{par hypothèse d'induction} \\
&= |\neg p|_) && \text{par définition } |\cdot|_)
\end{aligned}
$$

Rédaction d'une preuve par induction (2)

(2) Cas de la négation :

Soit $p \in \mathit{Form}$. Hypothèse d'induction : $|p|_(= |p|_)$.
À montrer : $|\neg p|_(= |\neg p|_)$.

Démonstration :

$$\begin{array}{rcll} |\neg p|_(& = & |p|_(& \text{par définition } |\cdot|_(\\ & = & |p|_) & \text{par hypothèse d'induction} \\ & = & |\neg p|_) & \text{par définition } |\cdot|_) \end{array}$$

Rédaction d'une preuve par induction (2)

(2) Cas de la négation :

Soit $p \in \mathit{Form}$. Hypothèse d'induction : $|p|_(= |p|_)$.
À montrer : $|\neg p|_(= |\neg p|_)$.

Démonstration :

$$\begin{array}{rcll} |\neg p|_(& = & |p|_(& \text{par définition } |\cdot|_(\\ & = & |p|_) & \text{par hypothèse d'induction} \\ & = & |\neg p|_) & \text{par définition } |\cdot|_) \end{array}$$

Rédaction d'une preuve par induction (2)

(2) Cas de la négation :

Soit $p \in Form$. Hypothèse d'induction : $|p|_(= |p|_)$.
À montrer : $|\neg p|_(= |\neg p|_)$.

Démonstration :

$$\begin{array}{rcll} |\neg p|_(& = & |p|_(& \text{par définition } |\cdot|_(\\ & = & |p|_) & \text{par hypothèse d'induction} \\ & = & |\neg p|_) & \text{par définition } |\cdot|_) \end{array}$$

Rédaction d'une preuve par induction (3)

(3) Cas de la conjonction :

Soient $p, q \in \mathit{Form}$.
Hypothèse d'induction : $|p|_(= |p|_)$ et $|q|_(= |q|_)$
À montrer : $|(p \wedge q)|_(= |(p \wedge q)|_)$

Démonstration :

$$
\begin{aligned}
|(p \wedge q)|_(&= 1 + |p|_(+ |q|_(&& \text{par définition } |\cdot|_(\\
&= |p|_) + |q|_) + 1 && \text{par hypothèse d'induction} \\
&= |(p \wedge q)|_) && \text{par définition } |\cdot|_)
\end{aligned}
$$

Rédaction d'une preuve par induction (3)

(3) Cas de la conjonction :

Soient $p, q \in Form$.
Hypothèse d'induction : $|p|_(= |p|_)$ et $|q|_(= |q|_)$
À montrer : $|(p \wedge q)|_(= |(p \wedge q)|_)$

Démonstration :

$$
\begin{array}{lcll}
|(p \wedge q)|_(& = & 1 + |p|_(+ |q|_(& \text{par définition } |\cdot|_(\\
 & = & |p|_) + |q|_) + 1 & \text{par hypothèse d'induction} \\
 & = & |(p \wedge q)|_) & \text{par définition } |\cdot|_)
\end{array}
$$

Rédaction d'une preuve par induction (3)

(3) Cas de la conjonction :

Soient $p, q \in Form$.
Hypothèse d'induction : $|p|_(= |p|_)$ et $|q|_(= |q|_)$
À montrer : $|(p \wedge q)|_(= |(p \wedge q)|_)$

Démonstration :

$$
\begin{array}{lcll}
|(p \wedge q)|_(& = & 1 + |p|_(+ |q|_(& \text{par définition } |\cdot|_(\\
 & = & |p|_) + |q|_) + 1 & \text{par hypothèse d'induction} \\
 & = & |(p \wedge q)|_) & \text{par définition } |\cdot|_)
\end{array}
$$

Rédaction d'une preuve par induction (3)

(3) Cas de la conjonction :

Soient $p, q \in Form$.
Hypothèse d'induction : $|p|_(= |p|_)$ et $|q|_(= |q|_)$
À montrer : $|(p \wedge q)|_(= |(p \wedge q)|_)$

Démonstration :

$$
\begin{array}{rcll}
|(p \wedge q)|_(& = & 1 + |p|_(+ |q|_(& \text{par définition } |\cdot|_(\\
 & = & |p|_) + |q|_) + 1 & \text{par hypothèse d'induction} \\
 & = & |(p \wedge q)|_) & \text{par définition } |\cdot|_)
\end{array}
$$

Rédaction d'une preuve par induction (3)

(3) Cas de la conjonction :

Soient $p, q \in Form$.
Hypothèse d'induction : $|p|_(= |p|_)$ et $|q|_(= |q|_)$
À montrer : $|(p \wedge q)|_(= |(p \wedge q)|_)$

Démonstration :

$$
\begin{array}{lcll}
|(p \wedge q)|_(& = & 1 + |p|_(+ |q|_(& \text{par définition } |\cdot|_(\\
& = & |p|_) + |q|_) + 1 & \text{par hypothèse d'induction} \\
& = & |(p \wedge q)|_) & \text{par définition } |\cdot|_)
\end{array}
$$

Rédaction d'une preuve par induction (4)

(4) Cas de la disjonction :

Soient $p, q \in Form$.
Hypothèse d'induction : $|p|_(= |p|_)$ et $|q|_(= |q|_)$
À montrer : $|(p \vee q)|_(= |(p \vee q)|_)$

Démonstration :

$$\begin{array}{rcll} |(p \vee q)|_(& = & 1 + |p|_(+ |q|_(& \text{par définition } |\cdot|_(\\ & = & |p|_) + |q|_) + 1 & \text{par hypothèse d'induction} \\ & = & |(p \vee q)|_) & \text{par définition } |\cdot|_) \end{array}$$

Rédaction d'une preuve par induction (4)

(4) Cas de la disjonction :

Soient $p, q \in Form$.
Hypothèse d'induction : $|p|_(= |p|_)$ et $|q|_(= |q|_)$
À montrer : $|(p \vee q)|_(= |(p \vee q)|_)$

Démonstration :

$$\begin{array}{rcll} |(p \vee q)|_(& = & 1 + |p|_(+ |q|_(& \text{par définition } |\cdot|_(\\ & = & |p|_) + |q|_) + 1 & \text{par hypothèse d'induction} \\ & = & |(p \vee q)|_) & \text{par définition } |\cdot|_) \end{array}$$

Rédaction d'une preuve par induction (4)

(4) Cas de la disjonction :

Soient $p, q \in \mathit{Form}$.
Hypothèse d'induction : $|p|_(= |p|_)$ et $|q|_(= |q|_)$
À montrer : $|(p \vee q)|_(= |(p \vee q)|_)$

Démonstration :

$$\begin{array}{rcll} |(p \vee q)|_(& = & 1 + |p|_(+ |q|_(& \text{par définition } |\cdot|_(\\ & = & |p|_) + |q|_) + 1 & \text{par hypothèse d'induction} \\ & = & |(p \vee q)|_) & \text{par définition } |\cdot|_) \end{array}$$

Rédaction d'une preuve par induction (4)

(4) Cas de la disjonction :

Soient $p, q \in Form$.
Hypothèse d'induction : $|p|_(= |p|_)$ et $|q|_(= |q|_)$
À montrer : $|(p \vee q)|_(= |(p \vee q)|_)$

Démonstration :

$$\begin{array}{rcll} |(p \vee q)|_(& = & 1 + |p|_(+ |q|_(& \text{par définition } |\cdot|_(\\ & = & |p|_) + |q|_) + 1 & \text{par hypothèse d'induction} \\ & = & |(p \vee q)|_) & \text{par définition } |\cdot|_) \end{array}$$

Rédaction d'une preuve par induction (4)

(4) Cas de la disjonction :

Soient $p, q \in Form$.
Hypothèse d'induction : $|p|_(= |p|_)$ et $|q|_(= |q|_)$
À montrer : $|(p \vee q)|_(= |(p \vee q)|_)$

Démonstration :

$$\begin{array}{lcll} |(p \vee q)|_(& = & 1 + |p|_(+ |q|_(& \text{par définition } |\cdot|_(\\ & = & |p|_) + |q|_) + 1 & \text{par hypothèse d'induction} \\ & = & |(p \vee q)|_) & \text{par définition } |\cdot|_) \end{array}$$

Remarques

- On utilise le théorème sur le principe des preuves par induction structurelle.
- Toujours dire clairement quelles sont les hypothèses, et qu'est-ce qu'il faut montrer.
- Justifier les étapes du raisonnement.
- Parfois (mais pas toujours !) ,le cas $\vee$ et le cas $\wedge$ sont très similaires.

Remarques

- On utilise le théorème sur le principe des preuves par induction structurelle.
- Toujours dire clairement quelles sont les hypothèses, et qu'est-ce qu'il faut montrer.
- Justifier les étapes du raisonnement.
- Parfois (mais pas toujours !) ,le cas $\vee$ et le cas $\wedge$ sont très similaires.

Remarques

- On utilise le théorème sur le principe des preuves par induction structurelle.
- Toujours dire clairement quelles sont les hypothèses, et qu'est-ce qu'il faut montrer.
- Justifier les étapes du raisonnement.
- Parfois (mais pas toujours !) ,le cas $\vee$ et le cas $\wedge$ sont très similaires.

Remarques

- On utilise le théorème sur le principe des preuves par induction structurelle.
- Toujours dire clairement quelles sont les hypothèses, et qu'est-ce qu'il faut montrer.
- Justifier les étapes du raisonnement.
- Parfois (mais pas toujours !) ,le cas $\vee$ et le cas $\wedge$ sont très similaires.

Retour au théorème

Théorème : Soit P une propriété sur les chaînes de caractères. Supposons que P vérifie les énoncés de clôture suivants :

- tout élément de V a la propriété P,
- si p satisfait P alors $\neg p$ satisfait P,
- si p et q satisfont P alors $(p \wedge q)$ satisfait P,
- si p et q satisfont P alors $(p \vee q)$ satisfait P,

Alors tous les éléments de *Form* satisfont P.

Démonstration

- Soit X l'ensemble de toutes les chaînes de caractères avec la propriété P.
- On remarque donc que l'ensemble X satisfait les propriétés de clôture des formules propositionnelles (par l'hypothèse du théorème).
- Donc $Form \subseteq X$ car $Form$ est le plus petit ensemble de chaînes de caractères qui satisfait ces propriétés.
- Autrement dit, tout élément de $Form$ a la propriété P.

Démonstration

- Soit X l'ensemble de toutes les chaînes de caractères avec la propriété P.
- On remarque donc que l'ensemble X satisfait les propriétés de clôture des formules propositionnelles (par l'hypothèse du théorème).
- Donc $Form \subseteq X$ car *Form* est le plus petit ensemble de chaînes de caractères qui satisfait ces propriétés.
- Autrement dit, tout élément de *Form* a la propriété P.

Démonstration

- Soit X l'ensemble de toutes les chaînes de caractères avec la propriété P.
- On remarque donc que l'ensemble X satisfait les propriétés de clôture des formules propositionnelles (par l'hypothèse du théorème).
- Donc $Form \subseteq X$ car $Form$ est le plus petit ensemble de chaînes de caractères qui satisfait ces propriétés.
- Autrement dit, tout élément de $Form$ a la propriété P.

Démonstration

- Soit X l'ensemble de toutes les chaînes de caractères avec la propriété P.
- On remarque donc que l'ensemble X satisfait les propriétés de clôture des formules propositionnelles (par l'hypothèse du théorème).
- Donc $\mathit{Form} \subseteq X$ car Form est le plus petit ensemble de chaînes de caractères qui satisfait ces propriétés.
- Autrement dit, tout élément de Form a la propriété P.

Définition d'une fonction

- Une méthode pour définir une fonction : par une expression close.
- Par exemple, la fonction qui envoie un argument, qui est un nombre naturel, vers l'argument incrémenté de 1 :

$$f(x) := x + 1$$

  La variable x est le paramètre formel de la fonction.
- Pour appliquer une telle fonction : remplacer dans l'expression le paramètre formel par le paramètre actuel, puis évaluer l'expression.
- Sur l'exemple :

$$f(5) = 5 + 1 = 6$$

Définition d'une fonction

- Une méthode pour définir une fonction : par une expression close.
- Par exemple, la fonction qui envoie un argument, qui est un nombre naturel, vers l'argument incrémenté de 1 :

$$f(x) := x + 1$$

La variable x est le paramètre formel de la fonction.

- Pour appliquer une telle fonction : remplacer dans l'expression le paramètre formel par le paramètre actuel, puis évaluer l'expression.
- Sur l'exemple :

$$f(5) = 5 + 1 = 6$$

Définition d'une fonction

- Une méthode pour définir une fonction : par une expression close.
- Par exemple, la fonction qui envoie un argument, qui est un nombre naturel, vers l'argument incrémenté de 1 :

$$f(x) := x + 1$$

  La variable x est le paramètre formel de la fonction.
- Pour appliquer une telle fonction : remplacer dans l'expression le paramètre formel par le paramètre actuel, puis évaluer l'expression.
- Sur l'exemple :

$$f(5) = 5 + 1 = 6$$

Définition d'une fonction

- Une méthode pour définir une fonction : par une expression close.
- Par exemple, la fonction qui envoie un argument, qui est un nombre naturel, vers l'argument incrémenté de 1 :

$$f(x) := x + 1$$

  La variable x est le paramètre formel de la fonction.
- Pour appliquer une telle fonction : remplacer dans l'expression le paramètre formel par le paramètre actuel, puis évaluer l'expression.
- Sur l'exemple :

$$f(5) = 5 + 1 = 6$$

Définition de fonctions sur *Form*

- Problème : comment définir une fonction sur un ensemble qui est défini par induction ?
- Exemple : comment définir la fonction qui donne la longueur d'une formule, ou le nombre de parenthèses ?
- En général, l'évaluation d'une telle fonction appliquée à une formule dépend de la structure de la formule.

Définition de fonctions sur *Form*

- Problème : comment définir une fonction sur un ensemble qui est défini par induction ?
- Exemple : comment définir la fonction qui donne la longueur d'une formule, ou le nombre de parenthèses ?
- En général, l'évaluation d'une telle fonction appliquée à une formule dépend de la structure de la formule.

Définition de fonctions sur *Form*

- Problème : comment définir une fonction sur un ensemble qui est défini par induction ?
- Exemple : comment définir la fonction qui donne la longueur d'une formule, ou le nombre de parenthèses ?
- En général, l'évaluation d'une telle fonction appliquée à une formule dépend de la structure de la formule.

Définition par récurrence

On peut définir une fonction avec domaine *Form* comme suit :

1. on donne le résultat de la fonction appliquée à un élément quelconque de V,
2. on donne le résultat de la fonction appliquée à une formule de la forme $\neg p$, sachant quel est le résultat de la fonction appliquée à p,
3. on donne le résultat de la fonction appliquée à une formule de la forme $(p \wedge q)$, sachant quel est le résultat de la fonction appliquée à p et le résultat de la fonction appliquée à q,
4. on donne le résultat de la fonction appliquée à une formule de la forme $(p \vee q)$, sachant quel est le résultat de la fonction appliquée à p et le résultat de la fonction appliquée à q.

Définition par récurrence

On peut définir une fonction avec domaine *Form* comme suit :

1. on donne le résultat de la fonction appliquée à un élément quelconque de V,
2. on donne le résultat de la fonction appliquée à une formule de la forme $\neg p$, sachant quel est le résultat de la fonction appliquée à p,
3. on donne le résultat de la fonction appliquée à une formule de la forme $(p \wedge q)$, sachant quel est le résultat de la fonction appliquée à p et le résultat de la fonction appliquée à q,
4. on donne le résultat de la fonction appliquée à une formule de la forme $(p \vee q)$, sachant quel est le résultat de la fonction appliquée à p et le résultat de la fonction appliquée à q.

Définition par récurrence

On peut définir une fonction avec domaine *Form* comme suit :

1. on donne le résultat de la fonction appliquée à un élément quelconque de V,
2. on donne le résultat de la fonction appliquée à une formule de la forme $\neg p$, sachant quel est le résultat de la fonction appliquée à p,
3. on donne le résultat de la fonction appliquée à une formule de la forme $(p \wedge q)$, sachant quel est le résultat de la fonction appliquée à p et le résultat de la fonction appliquée à q,
4. on donne le résultat de la fonction appliquée à une formule de la forme $(p \vee q)$, sachant quel est le résultat de la fonction appliquée à p et le résultat de la fonction appliquée à q.

Définition par récurrence

On peut définir une fonction avec domaine *Form* comme suit :

1. on donne le résultat de la fonction appliquée à un élément quelconque de V,
2. on donne le résultat de la fonction appliquée à une formule de la forme $\neg p$, sachant quel est le résultat de la fonction appliquée à p,
3. on donne le résultat de la fonction appliquée à une formule de la forme $(p \wedge q)$, sachant quel est le résultat de la fonction appliquée à p et le résultat de la fonction appliquée à q,
4. on donne le résultat de la fonction appliquée à une formule de la forme $(p \vee q)$, sachant quel est le résultat de la fonction appliquée à p et le résultat de la fonction appliquée à q.

Exemple récurrence : length

La fonction *length* est définie comme suit :

1. $length(x) = 1$ si $x \in V$
2. $length(\neg p) = 1 + length(p)$
3. $length((p \wedge q)) = 3 + length(p) + length(q)$
4. $length((p \vee q)) = 3 + length(p) + length(q)$

Exemple récurrence : length

Évaluation :

$$
\begin{aligned}
length((x_1 \wedge \neg x_2)) &= 3 + length(x_1) + length(\neg x_2) & (3)\\
&= 3 + 1 + length(\neg x_2) & (1)\\
&= 3 + 1 + 1 + length(x_2) & (2)\\
&= 3 + 1 + 1 + 1 & (1)\\
&= 6
\end{aligned}
$$

Exemple récurrence : length

Évaluation :

$$
\begin{aligned}
length((x_1 \wedge \neg x_2)) &= 3 + length(x_1) + length(\neg x_2) && (3)\\
&= 3 + 1 + length(\neg x_2) && (1)\\
&= 3 + 1 + 1 + length(x_2) && (2)\\
&= 3 + 1 + 1 + 1 && (1)\\
&= 6
\end{aligned}
$$

Exemple récurrence : length

Évaluation :

$$
\begin{aligned}
length((x_1 \wedge \neg x_2)) &= 3 + length(x_1) + length(\neg x_2) & (3)\\
&= 3 + 1 + length(\neg x_2) & (1)\\
&= 3 + 1 + 1 + length(x_2) & (2)\\
&= 3 + 1 + 1 + 1 & (1)\\
&= 6
\end{aligned}
$$

Exemple récurrence : length

Évaluation :

$$
\begin{array}{rcll}
length((x_1 \wedge \neg x_2)) & = & 3 + length(x_1) + length(\neg x_2) & (3) \\
 & = & 3 + 1 + length(\neg x_2) & (1) \\
 & = & 3 + 1 + 1 + length(x_2) & (2) \\
 & = & 3 + 1 + 1 + 1 & (1) \\
 & = & 6 &
\end{array}
$$

Exemple récurrence : length

Évaluation :

$$\begin{array}{lcll} length((x_1 \wedge \neg x_2)) & = & 3 + length(x_1) + length(\neg x_2) & (3) \\ & = & 3 + 1 + length(\neg x_2) & (1) \\ & = & 3 + 1 + 1 + length(x_2) & (2) \\ & = & 3 + 1 + 1 + 1 & (1) \\ & = & 6 & \end{array}$$

Exemple récurrence : length

Évaluation :

$$\begin{array}{lcll} length((x_1 \wedge \neg x_2)) & = & 3 + length(x_1) + length(\neg x_2) & (3) \\ & = & 3 + 1 + length(\neg x_2) & (1) \\ & = & 3 + 1 + 1 + length(x_2) & (2) \\ & = & 3 + 1 + 1 + 1 & (1) \\ & = & 6 & \end{array}$$

Exemple de récurrence : $\mathcal{V}$

Notre deuxième exemple est la fonction $\mathcal{V}$, définie comme suit :

1. $\mathcal{V}(x) = \{x\}$ si $x \in V$
2. $\mathcal{V}(\neg p) = \mathcal{V}(p)$
3. $\mathcal{V}((p \wedge q)) = \mathcal{V}(p) \cup \mathcal{V}(q)$
4. $\mathcal{V}((p \vee q)) = \mathcal{V}(p) \cup \mathcal{V}(q)$

$\mathcal{V}(p)$ est l'ensemble des variables de p.

Exemple de récurrence : $\mathcal{V}$

Notre deuxième exemple est la fonction $\mathcal{V}$, définie comme suit :

1. $\mathcal{V}(x) = \{x\}$ si $x \in V$
2. $\mathcal{V}(\neg p) = \mathcal{V}(p)$
3. $\mathcal{V}((p \wedge q)) = \mathcal{V}(p) \cup \mathcal{V}(q)$
4. $\mathcal{V}((p \vee q)) = \mathcal{V}(p) \cup \mathcal{V}(q)$

$\mathcal{V}(p)$ est l'ensemble des variables de p.

Exemple de récurrence : $\mathcal{V}$

Évaluation :

$$
\begin{array}{rcll}
\mathcal{V}((x_1 \wedge (x_2 \vee x_3))) & = & \mathcal{V}(x_1) \cup \mathcal{V}((x_2 \vee x_3)) & (3) \\
 & = & \{x_1\} \cup \mathcal{V}(x_2) \cup \mathcal{V}(x_3) & (1), (4) \\
 & = & \{x_1\} \cup \{x_2\} \cup \{x_3\} & (1), (1) \\
 & = & \{x_1, x_2, x_3\} &
\end{array}
$$

Exemple de récurrence : $\mathcal{V}$

Évaluation :

$$\begin{aligned}
\mathcal{V}((x_1 \wedge (x_2 \vee x_3))) &= \mathcal{V}(x_1) \cup \mathcal{V}((x_2 \vee x_3)) && (3) \\
&= \{x_1\} \cup \mathcal{V}(x_2) \cup \mathcal{V}(x_3) && (1), (4) \\
&= \{x_1\} \cup \{x_2\} \cup \{x_3\} && (1), (1) \\
&= \{x_1, x_2, x_3\}
\end{aligned}$$

Exemple de récurrence : $\mathcal{V}$

Évaluation :

$$\begin{array}{rcll} \mathcal{V}((x_1 \wedge (x_2 \vee x_3))) & = & \mathcal{V}(x_1) \cup \mathcal{V}((x_2 \vee x_3)) & (3) \\ & = & \{x_1\} \cup \mathcal{V}(x_2) \cup \mathcal{V}(x_3) & (1), (4) \\ & = & \{x_1\} \cup \{x_2\} \cup \{x_3\} & (1), (1) \\ & = & \{x_1, x_2, x_3\} & \end{array}$$

Exemple de récurrence : $\mathcal{V}$

Évaluation :

$$\begin{array}{rcll}
\mathcal{V}((x_1 \wedge (x_2 \vee x_3))) & = & \mathcal{V}(x_1) \cup \mathcal{V}((x_2 \vee x_3)) & (3) \\
& = & \{x_1\} \cup \mathcal{V}(x_2) \cup \mathcal{V}(x_3) & (1), (4) \\
& = & \{x_1\} \cup \{x_2\} \cup \{x_3\} & (1), (1) \\
& = & \{x_1, x_2, x_3\} &
\end{array}$$

Exemple de récurrence : $\mathcal{V}$

Évaluation :

$$\begin{array}{rcll}
\mathcal{V}((x_1 \wedge (x_2 \vee x_3))) & = & \mathcal{V}(x_1) \cup \mathcal{V}((x_2 \vee x_3)) & (3) \\
 & = & \{x_1\} \cup \mathcal{V}(x_2) \cup \mathcal{V}(x_3) & (1), (4) \\
 & = & \{x_1\} \cup \{x_2\} \cup \{x_3\} & (1), (1) \\
 & = & \{x_1, x_2, x_3\} &
\end{array}$$

Une subtilité

- Une fonction doit toujours associer à un argument donné un seul résultat. On doit donc assurer que la définition récursive d'une fonction garantit bien cette unicité du résultat.
- En principe, la même formule pourrait être construite de deux façon différentes. Dans ce cas, la définition de la fonction risque de donner deux valeurs différentes selon la construction considérée.
- Heureusement, cette difficulté n'existe pas pour notre définition des formules propositionnelles : *théorème de lecture unique* (démonstration omise).
- Mais attention dans le cas général des ensembles définis par induction.

Une subtilité

- Une fonction doit toujours associer à un argument donné un seul résultat. On doit donc assurer que la définition récursive d'une fonction garantit bien cette unicité du résultat.
- En principe, la même formule pourrait être construite de deux façon différentes. Dans ce cas, la définition de la fonction risque de donner deux valeurs différentes selon la construction considérée.
- Heureusement, cette difficulté n'existe pas pour notre définition des formules propositionnelles : *théorème de lecture unique* (démonstration omise).
- Mais attention dans le cas général des ensembles définis par induction.

Une subtilité

- Une fonction doit toujours associer à un argument donné un seul résultat. On doit donc assurer que la définition récursive d'une fonction garantit bien cette unicité du résultat.
- En principe, la même formule pourrait être construite de deux façon différentes. Dans ce cas, la définition de la fonction risque de donner deux valeurs différentes selon la construction considérée.
- Heureusement, cette difficulté n'existe pas pour notre définition des formules propositionnelles : *théorème de lecture unique* (démonstration omise).
- Mais attention dans le cas général des ensembles définis par induction.

Une subtilité

- Une fonction doit toujours associer à un argument donné un seul résultat. On doit donc assurer que la définition récursive d'une fonction garantit bien cette unicité du résultat.
- En principe, la même formule pourrait être construite de deux façon différentes. Dans ce cas, la définition de la fonction risque de donner deux valeurs différentes selon la construction considérée.
- Heureusement, cette difficulté n'existe pas pour notre définition des formules propositionnelles : *théorème de lecture unique* (démonstration omise).
- Mais attention dans le cas général des ensembles définis par induction.

Récurrence et induction

- Un ensemble peut être défini par induction: on dit comment construire un nouvel élément de l'ensemble à partir des éléments plus primitifs. Il y donc un sens « ascendant ».
- Les fonctions peuvent être définies par récurrence : on défini le résultat d'une fonction appliquée sur un argument composé en faisant référence au résultat de la fonction sur des arguments plus simples. Il y a donc un sens « descendant ».
- Finalement, une propriété de tous les éléments d'un ensemble qui est défini par induction est normalement démontrée par induction structurelle.

Récurrence et induction

- Un ensemble peut être défini par induction: on dit comment construire un nouvel élément de l'ensemble à partir des éléments plus primitifs. Il y donc un sens « ascendant ».
- Les fonctions peuvent être définies par récurrence : on défini le résultat d'une fonction appliquée sur un argument composé en faisant référence au résultat de la fonction sur des arguments plus simples. Il y a donc un sens « descendant ».
- Finalement, une propriété de tous les éléments d'un ensemble qui est défini par induction est normalement démontrée par induction structurelle.

Récurrence et induction

- Un ensemble peut être défini par induction: on dit comment construire un nouvel élément de l'ensemble à partir des éléments plus primitifs. Il y donc un sens « ascendant ».
- Les fonctions peuvent être définies par récurrence : on défini le résultat d'une fonction appliquée sur un argument composé en faisant référence au résultat de la fonction sur des arguments plus simples. Il y a donc un sens « descendant ».
- Finalement, une propriété de tous les éléments d'un ensemble qui est défini par induction est normalement démontrée par induction structurelle.