Chapitre 13 : Logique

26 mars 2024

1 Syntaxe et Sémantique

1.1 Syntaxe

On a vu précédemment :

- Les ensembles définis par inductions, sur lesquels on pouvait notamment faire des preuves par induction
- Les structures hiérarchiques, qui permettent notamment de représenter sous forme d'arbre les éléments d'un ensemble construit par induction (chaque noeud représentant un constructeur dont les fils sont les arguments).

Ces outils vont nous permettre de définir et étudier les *formules du calcul propositionnel*, qui sont l'ensemble des formules obtenues à partir de variables (et constantes) booléennes, et de leurs négations, conjonctions et disjonctions.

Définition

On définit de façon inductive l'ensemble $\mathcal{F}$ formules du calcul propositionnel sur un ensemble de symbole de variables $\mathcal{V}$:

- $\top, \bot \in \mathcal{F}$
- $\forall x \in \mathcal{V}, x \in \mathcal{F}$
- Soient ϕ_1 et $\phi_2 \in \mathcal{F}$, $\phi_1 \wedge \phi_2 \in \mathcal{F}$
- Soient ϕ_1 et $\phi_2 \in \mathcal{F}$, $\phi_1 \vee \phi_2 \in \mathcal{F}$
- Soit $\phi \in \mathcal{F}$, $\neg\phi \in \mathcal{F}$

Exercice : définir un type en OCaml pour représenter l'ensemble des formules logiques sur un ensemble potentiellement infini de symboles de variables.

Définition

Soit ϕ une formule du calcul propositionnel. Les variables utilisées dans la construction de ϕ sont appelées **variables libres**.

Les formules logiques sont des formules, et peuvent simplement être vues comme des séquences de caractères. On obtient *l'écriture linéaire* des formules logiques en ajoutant des parenthèses autour des formules non-réduites à un symbole de variable ou de constante lors de l'utilisation des constructeurs $\neg$, $\vee$ et $\wedge$.

L'ensemble des formules du calcul propositionnel étant défini par induction, une formule peut être représentée par un arbre dont les noeuds internes sont les constructeurs avec arguments et les feuilles sont les constructeurs sans arguments. L'arbre est appelé *arbre de syntaxe abstraite*.

Exercice : Donner l'arbre de syntaxe abstraite de la formule $(x_1 \wedge x_2) \vee (\neg x_3 \vee x_1)$.

Dans une formule ϕ donnée, une *sous-formule* de ϕ est la formule associée à un sous-arbre de l'arbre de syntaxe de ϕ.

Remarque : grâce au parenthésage dans l'écriture linéaire d'une formule, il existe une unique formule associée à une écriture linéaire (et donc un seul arbre de syntaxe abstraite) : la construction de la formule est non-ambiguë. Toute formule logique se décompose de manière unique en sous-formules.

Exercice : donner un exemple de formule pour lequel ce ne serait pas le cas sans parenthésage.

La *hauteur* et la *taille* d'une formule correspondent à la hauteur et à la taille de son arbre de syntaxe abstraite.

Exercice : donner une définition inductive de la taille et de la hauteur d'une en fonction des constructeurs.

Exercice : On pose $\phi_1 = x_1 \vee x_2$, $\phi_2 = x_3 \wedge \neg x_4$, $\phi_3 = phi_1 \vee x_3$ et $\phi_4 = phi_3 \wedge \phi_2$.

Donner l'écriture linéaire de ϕ_4, son arbre de syntaxe abstraite, sa taille et sa hauteur.

D'autres constructeurs peuvent être définis : par exemple, les constructeurs binaires d'implication $\rightarrow$ et d'équivalence $\leftrightarrow$.