Faculté des Sciences d'Orsay — Université Paris-Saclay
Licence Informatique, LDD Informatique, Mathématiques, semestre 5 — 2024–25
Eléments de logique pour l'informatique (Info 315) — 14 octobre 2024

Exercices de révision pour le partiel

Exercice 1 *Logique propositionnelle.*
Soit la formule A définie comme $((p \Rightarrow q \Rightarrow r) \Rightarrow r) \Rightarrow p \wedge q$.

1. Représenter la formule A sous forme d'arbre.
2. Donner la table de vérité de la formule A.
3. Dire si la formule est valide, satisfiable, insatisfiable ?
4. La formule $\neg A$ a-t-elle un modèle ? si oui donner un exemple.

Exercice 2 *Modélisation (partiel 2017)*

Dans cet exercice, on s'intéresse à modéliser un système de droit d'accès sur des ressources.

Les objets de la logique vont représenter des individus, des groupes d'individus, des ressources (fichiers, répertoires), des actions à réaliser (lire, écrire, supprimer,...). Les symboles de prédicat qui nous intéressent sont les suivants :

— $\texttt{action}(a)$: a est une action ;
— $\texttt{ressource}(r)$: r est une ressource ;
— $\texttt{groupe}(g)$: g est un groupe ;
— $\texttt{individu}(x)$: x est un individu ;
— $\texttt{dans}(x, g)$: l'individu x est dans le groupe g ;
— $\texttt{proprio}(x, r)$: l'individu x est le propriétaire de la ressource r ;
— $\texttt{droit}(g, a, r)$: le groupe g est autorisé à effectuer l'action a sur la ressource r ;
— $\texttt{peut}(x, a, r)$: l'individu x peut effectuer l'action a sur la ressource r ;
— $x = y$ les objets x et y sont égaux.

On introduit également trois constantes `Ecrire`, `Lire` et `Supp` qui représentent les actions d'écriture, de lecture et de suppression d'une ressource.

1. Traduire en langage naturel les formules suivantes :
   (a) $\forall x\, r, \texttt{peut}(x, \texttt{Ecrire}, r) \Rightarrow \texttt{peut}(x, \texttt{Supp}, r)$
   (b) $\forall r, \texttt{ressource}(r) \Rightarrow \exists x, \texttt{peut}(x, \texttt{Supp}, r)$
   (c) $\forall x, \exists a, \exists r, \texttt{ressource}(r) \wedge \texttt{action}(a) \wedge \neg\texttt{peut}(x, a, r)$
   (d) $\exists r, \texttt{ressource}(r) \wedge \forall a, \forall x, \neg\texttt{peut}(x, a, r)$
2. Exprimer comme des formules logiques les propriétés suivantes :
   (a) Toute ressource a au moins un propriétaire ;
   (b) Il existe une ressource qui a au moins deux propriétaires ;
   (c) Une personne peut effectuer une action sur une ressource si et seulement si elle en est propriétaire ou bien si elle appartient à un groupe qui a le droit d'effectuer cette action ;
   (d) Tout groupe qui a le droit d'écriture sur une ressource a aussi le droit de lecture.
3. Proposer une interprétation sur un domaine qui contient deux individus (A et B), un groupe G, deux ressources F et D et les trois actions `Ecrire`, `Lire` et `Supp` et qui vérifie l'ensemble des formules de la question précédente.
4. Est-il possible de trouver une interprétation qui vérifie toutes les conditions de la première question ?

Exercice 3 *Transformation de formules*

On se donne une signature avec pour les termes deux constantes a et b, et pour les formules un symbole de prédicat binaire R.

On veut transformer une formule close du calcul des prédicats sur la signature précédente en une formule close **sans quantificateur** qui est équivalente si on se place sur le domaine à deux élements $\mathcal{D} = \{a, b\}$. On rappelle que pour toute interprétation sur ce domaine, et pour une formule A quelconque on a $(\forall x, A) \Leftrightarrow (A[x \leftarrow a] \wedge A[x \leftarrow b])$ et $(\exists x, A) \Leftrightarrow (A[x \leftarrow a] \vee A[x \leftarrow b])$. La notation $A[x \leftarrow t]$ désigne la formule A dans laquelle on a remplacé les occurrences libres de la variable x par le terme t.

1. Appliquer la transformation à la formule $\forall x, \exists y, R(a, y) \wedge R(x, y)$ (notée P) et simplifier la formule obtenue.
2. La formule P est-elle valide ? satisfiable ? justifier votre réponse.
3. Définir par des équations récursives une fonction `subst` qui étant donnée une formule A du calcul des prédicats sur la signature ci-dessus, une variable x et un terme t, retourne la formule $A[x \leftarrow t]$.
4. Définir par des équations récursives une fonction `dom2` qui étant donnée une formule du calcul des prédicats sans variable libre calcule une formule close sans quantificateur, équivalente dans toute interprétation sur un domaine à deux éléments $D = \{a, b\}$ en utilisant la méthode rappelée ci-dessus et la fonction `subst`.

   Justifier la terminaison du calcul.

Exercice 4 On introduit le connecteur $\diamond$ (nor) dont la table de vérité est donnée par

x	y	$x \diamond y$
V	V	F
V	F	F
F	V	F
F	F	V

1. Donner une forme normale conjonctive et une forme normale disjonctive de la formule $x \diamond y$.
2. Donner les tables de vérité des formules $(x \diamond x)$, $((x \diamond x) \diamond x)$ et $(x \diamond y) \diamond (x \diamond y)$
3. Donner des formules équivalentes à $\top$, $\bot$, $\neg x$, $x \wedge y$, $x \vee y$ et $x \Rightarrow y$ qui n'utilisent que l'opérateur $\diamond$ et possiblement les variables x et y. On pourra justifier le résultat soit par des tables de vérité, soit en utilisant des équivalences avec les formules propositionnelles usuelles.
4. Définir par des équations récursives une fonction `nor` qui étant donnée une formule propositionnelle construite sur les connecteurs habituels, la transforme en une formule qui ne contient que des variables propositionnelles et le connecteur $\diamond$.
5. Donner le résultat de `nor`$(x \vee (y \vee z))$.