Tutorat logique : TD4

Université François Rabelais

Département informatique de Blois

Logique pour l'informatique

*
* *

Problème 1

Les énoncés sont indépendants.

1. Soit l'ensemble de formules :

$$\mathcal{F}_1 = \{\forall x \cdot (P(x) \Rightarrow Q(x)), \forall x \cdot (Q(x) \Rightarrow R(x))\}$$

   Et la formule $C = \forall x \cdot (P(x) \Rightarrow R(x))$. A-t-on $\mathcal{F}_1 \models C$?

2. Soit l'ensemble de formules :

$$\mathcal{F}_2 = \{\forall x \cdot (P(x) \Rightarrow \forall y \cdot P(y)), \forall x \cdot (P(x) \vee Q(x))\}$$

   Et la formule $C = \exists x \cdot (\neg Q(x) \Rightarrow \forall y \cdot P(y))$. A-t-on $\mathcal{F}_2 \models C$?

3. Soient les énoncés suivants :
   $\Psi_1 = \forall x \cdot \forall y \cdot \forall z \cdot (F(x,y) \wedge F(y,z)) \Rightarrow G(x,z))$
   $\Psi_2 = \forall x \cdot \exists y \cdot F(y,x)$
   $\Psi_3 = \neg \forall x \cdot \exists y \cdot G(y,x)$
   L'ensemble $\mathcal{F}_3 = \{\Psi_1, \Psi_2, \Psi_3\}$ est-il cohérent ?

Problème 2

Soient les énoncés suivants Φ_i du raisonnement R :

Φ_1 *: Pour tout crime, il y'a un quelqu'un qui l'a commis.*

Φ_2 *: Il y'a uniquement les gens malhonnêtes qui commettent des crimes.*

Φ_3 *: Ne sont arrêtés que les gens malhonnêtes.*

Φ_4 *: Les gens malhonnêtes arrêtés ne commettent pas de crimes.*

Φ_5 *: Il y'a des crimes.*

C *: Il y'a des gens malhonnêtes non arrêtés.*

On définit le langage du premier ordre $\mathcal{L} = \{Cr, M, A, C\}$: *Cr(x)* d'arité 1 vrai si x est un crime, *M(y)* d'arité 1 vrai si y est malhonnête, *A(y)* vrai si y est arrêté et *C(x,y)* d'arité 2 vrai si y a commis le crime x.

1. Représenter en logique des prédicats les énoncés Φ_i et C.

2. Donner la forme de Skolem correspondante aux énoncés Φ_i et C respectifs.

3. A-t-on $\bigcup_{\Phi_i \in R} \Phi_i \models C$? Le montrer par résolution.

Problème 3

1. On définit la suite de Fibonacci $(\mathcal{F}_n)_{n\in\mathbb{N}}$ telle que : $\begin{cases} \mathcal{F}_{n+2} & = \mathcal{F}_{n+1} + \mathcal{F}_n \\ \mathcal{F}_0 & = 0 \\ \mathcal{F}_1 & = 1 \end{cases}$

   Écrire en Prolog un programme `fib(N,R)` qui calcule le terme $\mathcal{F}_n =$ `R` pour un rang `N` donné.

   Le programme renverra par exemple :

```
[1] ?-fib(6, R)
R = 8
[2] ?-fib(6, 12)
Failure
```

2. Écrire en Prolog un programme `subst(N, O, l, l')` qui prend une liste `l` en paramètre et retourne une liste `l'` avec tous les éléments `O` remplacés par `N`.

   Le programme renverra :

```
[1] ?-subst(new, old, [a, old, b, c, old], R).
R = [a, new, b, c, new]
[2] ?-subst(0, 1, [1, 0, 0, 1], [0 ,0 ,0 ,0]).
Success
```

Problème 4

Soit p, un symbole de prédicat d'arité 2.

1. Écrire une formule ϕ qui traduit que p est est une relation symétrique, transitive et que tout élément a une image par la relation (i.e. tout élément x a au moins une image y telle que $p(x, y)$).
2. Montrer en utilisant la méthode de résolution que p est réflexive.

Rappel : Une relation R est symétrique si pour tout $(d, d') \in R$, on a $(d', d) \in R$. Elle est transitive si pour tout $(d, d') \in R$ et tout $(d', d'') \in R$, on a $(d, d'') \in R$. Enfin, elle est réflexive si tout couple $(d, d) \in R$.

Problème 5

Traduire les énoncés suivants en logique des prédicats :

1. Tout le monde a menti à quelqu'un dans sa vie.
2. Un chat est entré.
3. Bien que personne ne fasse de bruit, Jean-Yves n'arrive pas à se concentrer.
4. Quelqu'un cite un philosophe qui n'a rien écrit.
5. Il y'a [au moins] un exercice qu'aucun mathématicien ne sait résoudre.

Problème 6

On sait que la logique des prédicats du premier ordre forme une théorie *indécidable*, c'est-à-dire qu'il n'existe pas d'algorithme permettant de vérifier si une formule logique Φ est valide ou non.
On considère le raisonnement suivant :

$$\{\forall x \cdot (P(x) \Rightarrow Q(f(x))), \forall x \cdot (Q(x) \Rightarrow P(f(x))), P(A)\} \models \forall x \cdot P(x)$$

Peut-on conclure que ce raisonnement est valide ? Si non, essayer de montrer que la conclusion n'est pas une conséquence logique des hypothèses en exhibant un contre-exemple.