Université Paris 7 - Licence Année 2004–2005

TD de Maths pour l'Info n° 9

Calcul propositionnel, Résolution propositionnelle, Syntaxe du calcul des prédicats

Exercice 1 (Calcul propositionnel)

1. Un ensemble contradictoire de formules contient-il forcément une formule contradictoire ?
2. Montrez qu'un ensemble contradictoire de formules contient un sous-ensemble fini qui est lui-même contradictoire.
3. Montrez que si $\Delta \models A$, alors il existe un ensemble fini $\Gamma \subseteq \Delta$ tel que $\Gamma \models A$.
4. Montrez qu'un ensemble Δ de formules est contradictoire si et seulement si il existe une formule A telle que $\Delta \models A$ et $\Delta \models \neg A$.

Exercice 2 (Résolution propositionnelle)

1. En utilisant la méthode de résolution, montrer que la formule

$$((p \vee q \vee \neg s) \wedge (\neg p \vee r \vee \neg s) \wedge s \wedge \neg q) \to r$$

   est valide.
2. Un club privé écossais a pour règles :
   - Tout membre non écossais porte des chaussettes rouges
   - Tout membre porte un kilt ou ne porte pas de chaussettes rouges
   - Les membres mariés ne sortent pas le dimanche
   - Un membre sort le dimanche si et seulement si il est écossais
   - Tout membre qui porte un kilt est écossais et marié
   - Tout membre écossais porte un kilt

   Pourquoi n'y a-t-il personne dans le club ?

Exercice 3 (Résolution propositionnelle – Implémentation) En Caml, les formules du calcul propositionnel peuvent être représentées par le type de données suivant :

```
type formule =
  | Var of string               (* Variable propositionnelle *)
  | Non of formule              (* Négation *)
  | Et of formule * formule     (* Conjonction *)
  | Ou of formule * formule     (* Disjonction *)
  | Imp of formule * formule    (* Implication *)
  | Eqv of formule * formule ;; (* Équivalence logique *)
```

Avec cette représentation, la formule $(p \wedge q) \leftrightarrow (\neg p \to q)$ s'écrit :

```
Eqv(Et(Var "p", Var "q"), Imp(Non(Var "p"), Var "q"))
```

Chaque clause (disjonctive ou conjonctive) est représentée par un enregistrement contenant deux listes de littéraux : les littéraux positifs (champ `pos`) et les littéraux négatifs (champ `neg`) :

```
type clause = { pos : string list ; neg : string list } ;;
```

Écrire deux fonctions `fnc` et `fnd` de type `formule` $\rightarrow$ `clause list` calculant respectivement la forme normale conjonctive et la forme normale disjonctive d'une formule donnée, sous la forme d'une liste de clauses (qui représente leur conjonction ou leur disjonction suivant le type de forme normale auquel on s'intéresse).

Exercice 4 (Calcul des prédicats – Syntaxe) Soient :

- p, q deux symboles de prédicats binaires,
- r un symbole de prédicat unaire,
- f un symbole de fonction unaire,
- a un symbole de fonction 0-aire et
- g un symbole de fonction ternaire.

Soit la formule du calcul des prédicats suivante :

$$F \quad = \quad \exists x \; p(x, f(y)) \vee \neg \forall y \; q(y, g(a, z, h(z)))$$

1. Donner tous les termes apparaissant dans F
2. Donner toutes les sous-formules de F.

Exercice 5 (Calcul des prédicats – Formalisation)
En utilisant les symboles de prédicat suivants

$C(x)$	« x est un champignon »
$V(x)$	« x est violet »
$E(x)$	« x est vénéneux »
$Eq(x, y)$	« x est égal à y »

traduire les énoncés suivants dans le calcul des prédicats du premier ordre :

1. Tous les champignons violets sont vénéneux.
2. Aucun champignon violet n'est vénéneux.
3. Tout champignon est violet ou vénéneux.
4. Tout champignon est soit violet soit vénéneux, mais pas les deux ("ou" exclusif).
5. Tous les champignons violets, sauf un, sont vénéneux.
6. Il existe exactement deux champignons violets.