

## TP Logique n° 2. Assistant de preuve COQ.

### 1 Préliminaires

Ce sujet consiste à utiliser l'assistant de preuve COQ pour démontrer des théorèmes de la logique des prédicats.

1. Démarrez COQ avec : coqide

### 2 Démonstrations en calcul des prédicats

Nous utiliserons ici les règles fournies par COQ résumées dans le guide disponible séparément sur moodle. Télécharger depuis moodle le fichier `tp2.v` qui contient ces commandes ainsi que l'exemple détaillé par la suite.

1. Pour utiliser Coq pour prouver les théorèmes suivants, il est nécessaire d'effectuer les commandes :

```
(* Ouverture d'une section *)
Section LogiquePredicats.
(* Définition de 2 domaines pour les prédicats *)
Variable A B : Type.
```

2. Démontrez :

```
(* Formule du second ordre : Quantification des prédicats P et Q *)
Theorem Thm_8 : forall (P Q : A -> Prop),
  (forall x1 : A, (P x1) /\ (Q x1))
-> (forall x2 : A, (P x2)) /\ (forall x3 : A, (Q x3)).
```

3. Démontrez :

```
(* Formule du second ordre : Quantification du prédicat P *)
Theorem Thm_9 : forall (P : A -> B -> Prop),
  (exists x1 : A, forall y1 : B, (P x1 y1))
-> forall y2 : B, exists x2 : A, (P x2 y2).
```

4. Démontrez :

```
(* Formule du second ordre : Quantification des prédicats P et Q *)
Theorem Thm_10 : forall (P Q : A -> Prop),
  (exists x1 : A, (P x1) -> (Q x1))
-> (forall x2 : A, (P x2)) -> exists x3 : A, (Q x3).
```

5. Démontrez :

```
(* Type représentant l'humanité *)
Variable H : Type.
(* Prédicat définissant la paternité *)
Variable P : H -> H -> Prop.
(* Prédicat définissant la grand paternité paternelle *)
Variable GPP : H -> H -> Prop.

(* Formule du premier ordre : *)
Theorem Thm_11 : (forall (e : H), exists (p : H), (P e p))
  /\
  (forall (e : H), forall (gpp : H),
    ((GPP e gpp)
     <->
     exists (p : H), (P e p) /\ (P p gpp))
  )
->
(forall (e : H), exists (gpp : H), (GPP e gpp)).
```

6. Enfin, il est nécessaire de fermer la section :  
End LogiquePredicats.

## TP Logique n° 3. Atelier de vérification Why3

### 3 Préliminaires

Ce sujet consiste à utiliser l'atelier de vérification WHY3 pour démontrer des théorèmes de la logique des propositions et des prédicats. WHY3 repose sur un langage de modélisation combinant logique du premier ordre pour la spécification, programmation fonctionnelle et impérative pour l'implantation. WHY3 est une passerelle vers de nombreux outils de vérification partiellement (assistants de preuve) ou totalement (techniques SAT et SMT) automatisés.

1. Lancer la commande `why3 config detect` depuis la même fenêtre de commande pour configurer *Why3* en fonction des outils de preuve disponible dans votre environnement. Si vous avez déjà utilisé cet outil les années précédentes, il est préférable de supprimer le fichier de configuration avec la commande `rm $HOME/.why3.conf`.

### 4 Démonstrations en logique des propositions

1. Consulter le fichier `proposition.mlw` avec votre éditeur de texte préféré. Celui-ci comporte des :
  - déclarations de propositions en utilisant le mot clé `predicate` avec un nom commençant par une minuscule. Ce sont des prédicats d'arité 0 comme indiqué en cours magistral.
  - déclarations d'objectifs de preuve en utilisant le mot clé `goal` avec un nom commençant par une majuscule suivi de la formule qui doit être prouvée.
2. Démarrer WHY3 avec : `why3 ide proposition.mlw`. L'outil présente :
  - les différents objectifs de preuve (`goal` en Why3) de manière arborescente.
  - les outils de preuve automatique ou assistée qui peuvent être utilisés pour effectuer les preuves.
3. Sélectionner le premier but `Thm00`. Cliquer sur le bouton sélectionnant l'assistant de preuve `Coq`. Une tentative de preuve apparaît dans l'onglet associé au but. Sélectionner cette tentative puis cliquer sur le bouton `Edit` qui lancera `coqide`. Vous pouvez réaliser la preuve en `Coq` avec la tactique `tauto` et sauvegarder la preuve depuis `coqide`. Après avoir quitté `coqide`, vous pouvez jouer cette preuve en cliquant sur `Replay`.
4. Sélectionner le but suivant `Thm01`. Cliquer sur le bouton sélectionnant le prouveur automatique de type SMT `Alt-Ergo`. Celui-ci crée une tentative de preuve et réalise automatiquement cette preuve.
5. Sélectionner les autres buts et tenter de réaliser les preuves avec les différents outils disponibles.

### 5 Démonstrations en logique des prédicats

1. Consulter le fichier `predicat.mlw` avec votre éditeur de texte préféré. Celui-ci comporte des :
  - déclarations de types abstraits en utilisant le mot clé `type` avec un nom commençant par une minuscule.
  - déclarations de prédicats en précisant le type des paramètres de chaque prédicat.
  - déclarations d'objectifs de preuve.
2. Démarrer WHY3 avec : `why3 ide predicat.mlw`.
3. Sélectionner le premier but `Thm08`. Cliquer sur le bouton sélectionnant l'assistant de preuve `Coq`. Une tentative de preuve apparaît dans l'onglet associé au but. Sélectionner cette tentative puis cliquer sur le bouton `Edit` qui lancera `coqide`. Vous pouvez réaliser la preuve en `Coq` avec la tactique `firstorder` et sauvegarder la preuve depuis `coqide`. Après avoir quitté `coqide`, vous pouvez jouer cette preuve en cliquant sur `Replay`.
4. Sélectionner les autres buts et tenter de réaliser les preuves avec les différents outils automatiques disponibles.

## 6 Résolution d'une énigme logique

Les outils de preuve automatique en logique du premier ordre peuvent prouver qu'un objectif de preuve est valide mais ne peuvent pas prouver qu'il est invalide. Ils indiquent alors qu'ils n'ont pas réussi à faire la preuve sans assurance que l'objectif soit réellement invalide.

Pour utiliser ces outils pour résoudre une énigme logique, il est donc nécessaire de construire des objectifs de preuve dont la validité permet de répondre aux questions posées.

Utiliser l'atelier Why3 pour résoudre le problème traité dans la séance de Travaux Dirigés sur le point de vue sémantique en logique des propositions :

Trois personnes A,B et C, accusées d'un vol, déclarent respectivement :

— DA : B est coupable et C est innocent.

— DB : Si A est coupable alors C l'est aussi.

— DC : Je suis innocent mais au moins l'une des deux autres personnes est coupable.

Utiliser le formalisme du calcul des propositions pour modéliser les questions suivantes et donner la réponse pour chaque question :

1. Les trois déclarations sont-elles compatibles ?
2. L'un des témoignages peut-il se déduire des autres ? Lequel ?
3. Si tous sont innocents, lequel/lesquels a/ont menti ?
4. Si tous disent la vérité, qui est coupable ?
5. Si seuls les innocents disent la vérité, qui est innocent ?