Licence Informatique, semestre 5 2014–15
Eléments de logique pour l'informatique (Info 315) 6 janvier 2015
`http://www.lri.fr/~paulin/Logique`

Examen - 15 décembre 2014

L'examen dure 2 heures. L'énoncé est composé de 6 pages.
Le barème est indicatif. Toutes les réponses devront être clairement justifiées.
Le seul document autorisé est une page A4 manuscripte recto-verso. Le tableau résumant les règles du système G est donné à la fin du sujet.
Inscrivez votre nom sur chaque copie et numérotez-la.
Cacheter toutes les copies, recopier le numéro d'anonymat sur les intercalaires !

Exercice 1 *Preuves (4 points)*

On se place dans un langage avec deux symboles de prédicat unaires P et Q.
Soit la formule $A \stackrel{\text{def}}{=} ((\exists x, P(x)) \wedge (\forall x, Q(x))) \Rightarrow \exists x, (P(x) \wedge Q(x))$

Montrer que la formule A est valide de deux manières différentes :

1. faire une preuve dans le système G ;
2. faire une preuve par réfutation en utilisant la résolution.

Correction :

1.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\text{HYP}}{P(x), (\forall x, Q(x)) \vdash P(x)}\,(\forall_g) \quad \dfrac{\text{HYP}}{P(x), Q(x) \vdash Q(x)}}{P(x), (\forall x, Q(x)) \vdash Q(x)}}{P(x), (\forall x, Q(x)) \vdash P(x) \wedge Q(x)}\,(\wedge_d)}{P(x), (\forall x, Q(x)) \vdash \exists x, (P(x) \wedge Q(x))}\,(\exists_d)}{(\exists x, P(x)), (\forall x, Q(x)) \vdash \exists x, (P(x) \wedge Q(x))}\,(\exists_g)}{(\exists x, P(x)) \wedge (\forall x, Q(x)) \vdash \exists x, (P(x) \wedge Q(x))}\,(\wedge_g)}{\vdash ((\exists x, P(x)) \wedge (\forall x, Q(x))) \Rightarrow \exists x, (P(x) \wedge Q(x))}\,(\Rightarrow_d)$$

Il est essentiel d'appliquer la règle $\exists_g$ avant la règle $\exists_d$ (en partant du bas) afin de respecter la condition de la règle $\exists_g$ qui dit que la variable introduire ne peut pas être utilisée comme variable libre ailleurs dans le séquent.

2. On met la formule $\neg A$ en forme clausale

$$\neg A \equiv (\exists x, P(x)) \wedge (\forall x, Q(x)) \wedge (\forall x, \neg P(x) \vee \neg Q(x))$$

on skolémise en introduisant une constante a on obtient les trois clauses

$$C_1 \stackrel{\text{def}}{=} P(a) \qquad C_2 \stackrel{\text{def}}{=} Q(x) \qquad C_3 \stackrel{\text{def}}{=} \neg P(x) \vee \neg Q(x)$$

on résoud C_2 et C_3, on obtient la clause $C_4 \stackrel{\text{def}}{=} \neg P(x)$ que l'on résoud avec C_1 pour obtenir la clause vide.

Exercice 2 *Résolution (3 points)*

On considère un langage avec une constante e et trois symboles de fonction unaires a, b et c. On introduit également un prédicat binaire P et les formules suivantes :

$$A_0 \stackrel{\text{def}}{=} P(e,e) \qquad A_1 \stackrel{\text{def}}{=} \forall xy, P(x,y) \Rightarrow P(b(b(x)), b(y))$$
$$A_2 \stackrel{\text{def}}{=} \forall xy, P(x,y) \Rightarrow P(a(b(x)), b(a(y))) \qquad A_3 \stackrel{\text{def}}{=} \forall xy, P(x,y) \Rightarrow P(c(x), b(c(y)))$$

1. Mettre les formules A_0, A_1, A_2, A_3 en forme clausale.

   Correction :

   $$C_0 \stackrel{\text{def}}{=} P(e,e) \qquad C_1 \stackrel{\text{def}}{=} \neg P(x,y) \vee P(b(b(x)), b(y))$$
   $$C_2 \stackrel{\text{def}}{=} \neg P(x,y) \vee P(a(b(x)), b(a(y))) \qquad C_3 \stackrel{\text{def}}{=} \neg P(x,y) \vee P(c(x), b(c(y)))$$

2. A l'aide de la résolution montrer que $A_0, A_1, A_2, A_3 \models \exists z, P(b(z), b(z))$.

   On prendra soin de préciser pour chaque étape de résolution les clauses utilisées ainsi que les substitutions.

   Correction : *On met la formule $\neg\exists x, P(b(x), b(x))$ en forme clausale, on obtient*

   $$C_4 \stackrel{\text{def}}{=} \neg P(b(z), b(z))$$

   — *C_4 ne peut être résolue que avec C_1, le problème d'unification est $b(z) \stackrel{?}{=} b(b(x)) \stackrel{?}{=} b(y)$ ce qui donne après simplification la substitution $y \leftarrow b(x), z \leftarrow b(x)$ et la clause résultante $C_5 \stackrel{\text{def}}{=} \neg P(x, b(x))$ que l'on renomme en $\neg P(z, b(z))$.*
   — *C_5 peut être résolue avec C_1, C_2 ou C_3, on résoud avec C_3 le problème d'unification est $z \stackrel{?}{=} c(x)$ et $b(z) \stackrel{?}{=} b(c(y))$ ce qui donne après remplacement de z dans la deuxième équation $b(c(x)) \stackrel{?}{=} b(c(y))$ et donc la substitution $y \leftarrow x, z \leftarrow b(x)$ et la clause résultante $C_6 \stackrel{\text{def}}{=} \neg P(x,x)$.*
   — *C_6 peut se résoudre avec C_0 et on obtient la clause vide.*

Exercice 3 *Calcul des prédicats (3 points)*

On se place dans un langage avec un symbole de prédicat unaire P. Soit la formule $A \stackrel{\text{def}}{=} \exists x, (P(x) \Rightarrow \forall y, P(y))$.

1. Donner la forme de Skolem de A, on appelle B cette formule.
2. La formule A est-elle valide ? justifier la réponse.
3. Peut-on en déduire que la formule B est valide ? justifier la réponse.
4. On introduit un symbole de prédicat binaire R et la formule $C \stackrel{\text{def}}{=} \exists x, ((\exists z, R(x,z)) \Rightarrow \forall y, \exists z, R(y,z))$
   (a) Quel est le lien entre la formule A et la formule C ?
   (b) Peut-on en déduire que C est valide ?

Correction :

1. *$A \equiv \exists x, (\neg P(x) \vee \forall y, P(y))$ on introduit une constante a pour le connecteur existentiel et on obtient la formule $\forall y, \neg P(a) \vee P(y)$*

2. *on peut procéder par équivalence* $A \equiv \exists x, (\neg P(x) \vee \forall y, P(y)) \equiv (\exists x, (\neg P(x)) \vee \forall y, P(y) \equiv \neg(\forall x, P(x)) \vee \forall y, P(y)$ *qui est de la forme* $\neg C \vee C$ *donc toujours vrai.*
3. *On ne peut pas en déduire que* B *est valide (on a juste* $B \models A$*, mais pas l'inverse en général).* B *n'est pas valide, il suffit de prendre une interprétation avec deux éléments* $\{a, b\}$ *et* $P(a)$ *vrai et* $P(b)$ *faux.*
4. *La formule* C *est la formule* A *dans laquelle le prédicat* $P(x)$ *est remplacé par la formule* $\exists z, R(x, z)$*. La formule* A *étant valide, elle est vraie pour toute interprétation de* P *et donc* C *est aussi valide.*

Exercice 4 *Démonstration automatique (2 points)*
On suppose que l'on dispose d'une fonction `forme-clausale` qui étant donnée une formule F renvoie un ensemble de clauses équivalent à la forme de Skolem de F ainsi qu'une fonction `refut` qui étant donné un ensemble de clauses E, renvoie vrai si la clause vide est dérivable à partir de l'ensemble de clauses E.

1. Soit A et B deux formules, en utilisant les fonctions `forme-clausale` et `refut` ainsi que les opérations sur les formules et les opérations ensemblistes usuelles, décrire un algorithme qui répond vrai si B est conséquence logique de A. On pourra utiliser n'importe quel pseudo-langage pour décrire l'algorithme.
2. Est-il possible de construire un algorithme qui étant données deux formules A et B du calcul des prédicats réponde non si B n'est pas conséquence logique de A ? Justifier votre réponse en quelques lignes.

Correction :

1. B *est conséquence logique de* A *si et seulement si l'ensemble* $\{A, \neg B\}$ *est insatisfiable. Il suffit donc de mettre la formule* $A \wedge \neg B$ *en forme clausale puis d'appeler la fonction de réfutation.*

$$\texttt{consequence}(A, B) = \texttt{refut}(\texttt{forme-clausale}(A \wedge \neg B))$$

2. *Non ce n'est pas possible car le problème de validité dans le calcul des prédicats n'est pas décidable. Il existe un algorithme qui s'arrête et répond oui lorsque* B *est conséquence logique de* A*, si on avait une algorithme qui réponde non lorsque* B *n'est pas conséquence logique, il suffirait de faire tourner les deux algorthmes ensemble, au moins un des deux répondrait et on aurait une procédure de décision pour la validité des formules du calcul des prédicats.*

Exercice 5 *Modélisation (8 points)*
Dans cet exercice, on s'intéresse à modéliser un système de droit d'accès inspiré de celui des systèmes de fichier Unix.

Les questions sont largement indépendantes : il n'est pas nécessaire d'avoir résolu une question pour traiter les suivantes.

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.

1. Traduire en langage naturel les formules suivantes :
   (a) $\forall x\, r, \texttt{peut}(x, \texttt{Ecrire}, r) \Rightarrow \texttt{peut}(x, \texttt{Lire}, r)$
   `Ecrire` et `Lire` sont deux constantes représentant les actions d'écriture et de lecture d'une ressource.
   (b) $\exists x, \forall a\, r, \texttt{ressource}(r) \Rightarrow \texttt{action}(a) \Rightarrow \texttt{peut}(x, a, r)$

   Dans la suite ces conditions seront notées 1a et 1b.

   Correction :

   (a) Toute personne qui peut écrire sur une ressource peut aussi la lire

   (b) Il existe une personne qui peut effectuer toutes les actions sur toutes les ressources (un administrateur)

2. Exprimer comme des formules logiques les propriétés suivantes :
   (a) Le propriétaire d'une ressource peut effectuer toutes les actions sur cette ressource ;
   (b) Toute ressource a un et un seul propriétaire ;
   (c) Aucun groupe n'est vide et toute personne appartient à (au moins) un groupe.
   (d) Lorsqu'un groupe est autorisé à effectuer une action sur une ressource alors tous les membres de ce groupe peuvent effectuer l'action.

   Dans la suite ces conditions seront notées 2a, 2b, 2c et 2d.

   Correction :

   (a) $\forall xar, \texttt{proprio}(x, r) \wedge \texttt{action}(a) \Rightarrow \texttt{peut}(x, a, r)$

   (b) $\forall r, \texttt{ressource}(r) \Rightarrow \exists x, \texttt{proprio}(x, r) \wedge \forall y, \texttt{proprio}(y, r) \Rightarrow x = y$

   (c) $(\forall g, \texttt{groupe}(g) \Rightarrow \exists x, \texttt{dans}(x, g)) \wedge (\forall x, \texttt{individu}(x) \Rightarrow \exists g, \texttt{dans}(x, g))$

   (d) $\forall r\, g\, a\, x, \texttt{droit}(x, g, r) \wedge \texttt{dans}(x, g) \Rightarrow \texttt{peut}(x, a, r)$

3. Soit l'interprétation I sur le domaine

$$D \stackrel{\text{def}}{=} \{\texttt{Alice}, \texttt{Bob}, \texttt{Eve}, \texttt{Amis}, \texttt{Autres}, \texttt{Secret}, \texttt{Public}, \texttt{Lire}, \texttt{Ecrire}\}$$

   avec comme interprétation des prédicats les relations suivantes :
   — $\texttt{action}_I = \{\texttt{Lire}, \texttt{Ecrire}\}$
   — $\texttt{ressource}_I = \{\texttt{Secret}, \texttt{Public}\}$
   — $\texttt{groupe}_I = \{\texttt{Amis}, \texttt{Autres}\}$
   — $\texttt{individu}_I = \{\texttt{Alice}, \texttt{Bob}, \texttt{Eve}\}$
   — $\texttt{dans}_I = \{(\texttt{Alice}, \texttt{Amis}), (\texttt{Bob}, \texttt{Amis}), (\texttt{Eve}, \texttt{Autres})\}$
   — $\texttt{proprio}_I = \{(\texttt{Alice}, \texttt{Secret}), (\texttt{Bob}, \texttt{Public})\}$

— $\texttt{droit}_I = \{(\texttt{Amis}, \texttt{Lire}, \texttt{Secret}), (\texttt{Amis}, \texttt{Lire}, \texttt{Public}), (\texttt{Autres}, \texttt{Lire}, \texttt{Public})\}$

(a) Les conditions 2b et 2c sont-elles vérifiées dans l'interprétation I ?

(b) Trouver la plus petite relation pour interpréter le symbole de prédicat `peut` afin de vérifier les conditions 2a et 2d. On pourra représenter la réponse en complétant le tableau ci-dessous.

	(Lire,Secret)	(Ecrire,Secret)	(Lire,Public)	(Ecrire,Public)
Alice				
Bob				
Eve				

Si on met une croix sur la ligne `Alice` dans la colonne (`Lire`, `Secret`) cela signifie que `Alice` peut lire la ressource `Secret` dans l'interprétation I. Chercher la plus petite relation, revient à mettre un minimum de croix tout en assurant que les conditions 2a et 2d restent vraies.

(c) Les conditions 1a et 1b sont-elles vérifiées dans cette interprétation ? sinon proposer une autre interprétation du symbole de prédicat `peut` qui rend vraies toutes les conditions 1a, 1b, 2a et 2d.

(d) On suppose que l'interprétation I définit $\texttt{peut}_I$ de telle manière que les quatre propriétés 1a, 1b, 2a et 2d sont vraies. Peut-on en déduire que, dans cette interprétation, `Alice` peut écrire la ressource `Public` ? qu'elle ne peut pas écrire la ressource `Public` ?

Correction :

(a) Oui les conditions sont vérifiées : les ressources ont un seul propriétaire, chaque personne est dans un groupe et aucun groupe n'est vide.

(b) ***Alice*** *est propriétaire de* ***Secret*** *donc elle peut lire et écrire cette ressource ;*
Bob *est propriétaire de* ***Public*** *donc il peut lire et écrire cette ressource ;*
Alice *et* ***Bob*** *sont dans le groupe* ***Amis*** *donc ils peuvent lire les ressources* ***Secret*** *et* ***Public*** *;*
Eve *est dans le groupe* ***Autres*** *donc elle peut lire la ressource* ***Public****.*
Pour avoir la plus petite relation on suppose que ce sont les seuls cas pour lesquels ***peut*** *est vérifié*
Sous forme de tableau cela donne :

	(Lire,Secret)	*(Ecrire,Secret)*	*(Lire,Public)*	*(Ecrire,Public)*
Alice	X	X	X	
Bob	X		X	X
Eve			X	

(c) La condition 1a est bien vérifiée : ceux qui peuvent lire peuvent aussi écrire, par contre la condition 1b ne l'est pas : il n'y a personne qui a tous les droits. On peut changer le modèle par exemple en ajoutant le droit d'écrire ***Public*** *à* ***Alice*** *ou bien en ajoutant le droit d'écrire* ***Secret*** *à* ***Bob****.*

(d) Il existe une interprétation qui valide toutes les conditions mais dans laquelle ***Alice*** *peut écrire* ***Public*** *et il en existe une dans laquelle* ***Alice*** *ne peut pas écrire* ***Public****, donc on ne peut ni en déduire qu'Alice peut écrire la ressource public ni qu'elle ne le peut pas.*

4. Proposer une formule logique qui assure que les seules autorisations données sont celles correspondant aux conditions 2a et 2d, c'est-à-dire qu'on peut effectuer une action si on est propriétaire de la ressource ou si on est membre d'un groupe qui a le droit d'effectuer l'action sur la ressource.

   Correction : *On peut ajouter la formule*

   $$\forall x\, a\, r, \texttt{peut}(x, a, r) \Rightarrow \texttt{proprio}(x, r) \vee \exists g, \texttt{dans}(x, g) \wedge \texttt{droit}(g, a, r)$$

5. On veut changer le système de droit d'accès en regroupant les ressources en catégories et en déclarant les droits au niveau des catégories au lieu de le faire au niveau d'une ressource. C'est-à-dire que dans le prédicat $\texttt{droit}(g, a, c)$ on aura g un groupe, a une action et c une catégorie.

   Proposer une extension du langage pour traiter ce nouveau système et exprimer l'analogue de la propriété 2d dans ce nouveau système : c'est-à-dire qu'un individu peut effectuer une action sur une ressource lorsqu'il est dans un groupe qui a le droit de faire l'action sur une catégorie à laquelle l'objet appartient.

   Correction : *On ajoute un nouveau prédicat unaire* `categorie` *et un nouveau prédicat binaire* $\texttt{dans-cat}(r, c)$ *qui est vrai lorsque la ressource r est dans la catégorie c. L'analogue de la propriété 2d s'exprime alors*

   $$\forall x\, a\, r\, c\, g, \texttt{droit}(g, a, c) \wedge \texttt{dans}(x, g) \wedge \texttt{dans-cat}(r, c) \Rightarrow \texttt{peut}(x, a, r)$$

Rappel des règles logiques

hypothèse	$(\textsc{Hyp})\dfrac{}{A, \Gamma \vdash \Delta, A}$	
	gauche	droite
$\bot$	$\dfrac{}{\bot, \Gamma \vdash \Delta}$	$\dfrac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, \bot}$
$\top$	$\dfrac{\Gamma \vdash \Delta}{\top, \Gamma \vdash \Delta}$	$\dfrac{}{\Gamma \vdash \Delta, \top}$
$\neg$	$\dfrac{\Gamma \vdash \Delta, A}{\neg A, \Gamma \vdash \Delta}$	$\dfrac{A, \Gamma \vdash \Delta}{\Gamma \vdash \Delta, \neg A}$
$\wedge$	$\dfrac{A, B, \Gamma \vdash \Delta}{A \wedge B, \Gamma \vdash \Delta}$	$\dfrac{\Gamma \vdash \Delta, A \quad \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \wedge B}$
$\vee$	$\dfrac{A, \Gamma \vdash \Delta \quad B, \Gamma \vdash \Delta}{A \vee B, \Gamma \vdash \Delta}$	$\dfrac{\Gamma \vdash \Delta, A, B}{\Gamma \vdash \Delta, A \vee B}$
$\Rightarrow$	$\dfrac{\Gamma \vdash \Delta, A \quad B, \Gamma \vdash \Delta}{A \Rightarrow B, \Gamma \vdash \Delta}$	$\dfrac{A, \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \Rightarrow B}$
$\forall$	$\dfrac{P[x \leftarrow t], (\forall x, P), \Gamma \vdash \Delta}{(\forall x, P), \Gamma \vdash \Delta}$	$\dfrac{\Gamma \vdash \Delta, P \qquad x \notin \mathsf{Vl}(\Gamma, \Delta)}{\Gamma \vdash \Delta, (\forall x, P)}$
$\exists$	$\dfrac{P, \Gamma \vdash \Delta \qquad x \notin \mathsf{Vl}(\Gamma, \Delta)}{(\exists x, P), \Gamma \vdash \Delta}$	$\dfrac{\Gamma \vdash \Delta, (\exists x, P), P[x \leftarrow t]}{\Gamma \vdash \Delta, (\exists x, P)}$