Examen - 20 décembre 2012

L'examen dure 3 heures. L'énoncé est composé de 8 pages, le barème est indicatif. Toutes les réponses devront être clairement justifiées. Le seul document autorisé est une page A4 manuscripte recto-verso. Inscrivez votre nom sur chaque copie et numérotez-la. **Cacheter toutes les copies !**

Exercice 1 *Modélisation (4 points).*
On se place dans un langage du premier ordre modélisant les entiers qui utilise les symboles suivants :

- les constantes 0, 1 ;
- les symboles de fonction binaires $+$ et $*$ qui représentent l'addition et la multiplication et seront notés de manière usuelle $x + y$ et $x * y$;
- les symboles de prédicats unaires $\texttt{Pair}(x)$ et $\texttt{Prem}(x)$ représentant respectivement le fait que x est un nombre pair et x est un nombre premier (on rappelle qu'un nombre est premier s'il est strictement plus grand que 1 et n'est divisible que par 1 et par lui-même).
- les symboles de prédicats binaires $\texttt{Div}(y, x)$ qui représente le fait que y divise x, $x = y$ qui représente que x est égal à y et $x \leq y$ qui représente que x est inférieur ou égal à y.

1. Formaliser les énoncés suivants :
   (a) Il existe un entier plus petit ou égal à tous les autres.
   (b) Il n'existe pas d'entier plus grand ou égal à tous les autres, mais pour tout entier il en existe un qui est strictement plus grand.
   (c) Tout nombre entier pair est égal à la somme de deux nombres entiers premiers.
   (d) L'ensemble des entiers premiers est non borné.
2. Expliquer par des phrases le sens de chacune des formules suivantes et dire si elles sont vérifiées dans le modèle des entiers :
   (a) $\forall xy, (\texttt{Pair}(x) \land \texttt{Pair}(y) \Rightarrow \texttt{Pair}(x + y))$
   (b) $\forall xy, \exists z, (\texttt{Div}(x, z) \land \texttt{Div}(z, y))$
3. Pour chacun des prédicats suivants, donner une formule équivalente qui n'utilise que les symboles de constantes 0 et 1, les fonctions $+$ et $*$ et la relation d'égalité.
   (a) $\texttt{Pair}(x)$
   (b) $\texttt{Div}(y, x)$
   (c) $\texttt{Prem}(x)$ (on pourra utiliser le prédicat $\texttt{Div}$).

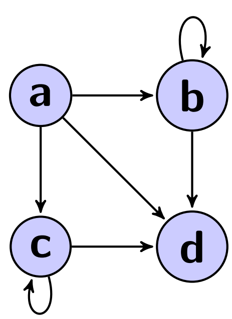
Correction :

1. *(a)* $\exists n, \forall m, n \leq m$
(b) $\neg(\exists n, \forall m, m \leq n) \land \forall m, \exists n, (m \leq n \land \neg n = m)$
(c) $\forall n, (\mathit{Pair}(n) \Rightarrow \exists p, \exists q, (n = p + q \land \mathit{Prem}(p) \land \mathit{Prem}(q)))$
(d) $\forall n, \exists p, (\mathit{Prem}(p) \land n \leq p)$

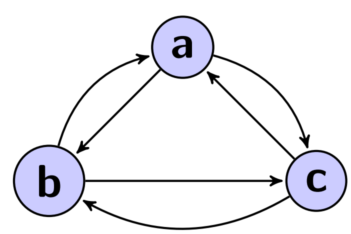
2. (a) *La somme de deux entiers pairs est pair, ce qui est vrai dans le modèle des entiers.*
   (b) *Pour tout entiers x et u, il existe z tel que x divise z et z divise y. Cette propriété est fausse dans le modèle des entiers, en effet on aurait alors que x divise y et il suffit de prendre $x = 2$ et $y = 3$ pour que la propriété soit fausse.*
3. (a) $\texttt{Pair}(x) \equiv \exists y, x = y + y$
   (b) $\texttt{Div}(y, x) \equiv \exists z, x = y * z$
   (c) $\texttt{Prem}(x) \equiv (1 + 1 \leq x \wedge \forall y, (\texttt{Div}(y, x) \Rightarrow (x = y \vee y = 1)))$

Exercice 2 *Modèles et Résolution (4 points).*
On se place dans un langage avec un symbole de prédicat binaire R. Soient les quatre formules de la logique du premier ordre suivantes :
- F_1 : $\forall x, ((\exists y, \neg R(x, y)) \Rightarrow \exists y, (R(x, y) \wedge R(y, x)))$
- F_2 : $\forall x, \exists y, (R(x, y) \vee R(y, x))$
- F_3 : $\forall xyz, (R(x, y) \wedge R(y, z) \Rightarrow R(x, z))$
- F_4 : $\exists x, R(x, x)$

1. On se donne des interprétations de la relation R sous forme de graphes. Le domaine est l'ensemble des sommets du graphe et on a une arête du sommet x au sommet y exactement lorsque la relation $R(x, y)$ est vérifiée dans l'interprétation. Dans chacune des deux interprétations suivantes :
   - donner la liste des couples (x, y) tels que $R(x, y)$ est vraie dans l'interprétation.
   - Dire lesquelles des formules F_1, F_2, F_3, F_4 précédentes sont vraies et lesquelles sont fausses. Justifier votre réponse.



modèle (A)



modèle (B)

2. Montrez **en utilisant la méthode de résolution** que
   (a) la formule F_2 est conséquence logique de la formule F_1 ;
   (b) la formule F_4 est conséquence des deux formules F_1 et F_3.

   Vous prendrez soin d'expliquer et de détailler toutes les étapes de votre démonstration (c'est-à-dire la mise en forme clausale et les étapes de la réfutation).
3. A l'aide d'une variante du modèle A, montrer que F_4 n'est pas conséquence logique de F_2 et F_3.

Correction :

1. (a) *Dans le premier modèle on a $R_A = \{(a, b), (a, c), (a, d), (b, b), (b, d), (c, c), (c, d)\}$ dans le second $R_B = \{(a, b), (a, c), (b, a), (b, c), (c, a), (c, b)\}$.*

(b) *La formule F_1 dit que si un point x n'est pas relié à tous les autres points alors il est lié à un point y dans les deux sens. La formule F_1 est fausse dans le modèle A (prendre par exemple le point a pour lequel il n'y a pas de x tel que $R(x, a)$. La formule F_1 est vraie dans le modèle B.*

La formule F_2 dit que tout point a soit un successeur soit un prédécesseur. Il est vrai dans les deux modèles.

La formule F_3 est la transitivité de la relation. C'est le cas du modèle A mais pas du modèle B car on a une flèche de a à b et de b à a mais pas de flèche de a à a.

La formule F_4 dit qu'il y a une boucle sur l'un des sommets et est vraie dans le modèle A mais fausse dans le modèle B.

2. (a) *Mise en forme clausale de $\{F_1, \neg F_2\}$:*
F_1 : $C_1 = R(x, y) \vee R(f(x), x)$ *et* $C_2 = R(x, y) \vee R(x, f(x))$
$\neg F_2$: $C_3 = \neg R(a, y)$ *et* $C_4 = \neg R(y, a)$.
On renomme C_3 en $C_3' = \neg R(a, z)$, on effectue la résolution entre C_2 et C_3' sur le litteral $R(x, f(x))$ avec la substitution $\{x \leftarrow a; z \leftarrow f(a)\}$ on obtient la clause $C_5 = R(a, y)$, il suffit de la recombiner avec C_3 pour aboutir à une contradiction.

   (b) *Mise en forme clausale de $\{F_1, F_3, \neg F_4\}$:*
F_1 : $C_1 = R(x, y) \vee R(f(x), x)$ *et* $C_2 = R(x, y) \vee R(x, f(x))$
F_3 : $C_4 = \neg R(x, y) \vee \neg R(y, z) \vee R(x, z)$
$\neg F_4$: $C_5 = \neg R(x, x)$
Réfutation : On fait la résolution entre C_1 et C_5 en utilisant la substitution $\{y \leftarrow x\}$ on obtient la clause $C_6 = R(f(x), x)$. On fait la résolution entre C_2 et C_5 en utilisant la substitution $\{y \leftarrow x\}$ on obtient la clause $C_7 = R(x, f(x))$. On fait la résolution entre C_4, C_6 puis C_7 et on obtient $C_8 = R(x, x)$. En combinant avec C_5 on aboutit à la clause vide.

3. *On prend un modèle A' analogue à A mais dans lequel on retire les deux boucles sur b et c. Ce modèle rend vraies les formules F_2 et F_3, par contre, dans ce modèle, F_4 est fausse.*

Exercice 3 *Syntaxe du calcul des prédicats (4 points).*
On se donne une signature avec deux constantes a et b, un symbole de fonction unaire f et deux symboles de prédicats P d'arité 1 et Q d'arité 2.

1. Pour les formules suivantes, dire si elles sont bien formées syntaxiquement (c'est-à-dire qu'elles respectent les contraintes d'arité de la signature). Si oui, donner la forme arborescente des formules en indiquant pour chaque variable si elle est libre ou liée. Si non expliquer pourquoi.
   (a) $\forall x, (P(x) \Rightarrow \exists y, Q(x, f(y)))$
   (b) $(\forall x, P(x)) \Rightarrow \exists y, Q(x, f(y))$
   (c) $\forall x, (P(x) \Rightarrow \exists y, Q(f(x, y)))$
2. On considère des formules du calcul des prédicats sur la signature donnée qui utilisent les connecteurs $\wedge$, $\vee$, $\neg$, $\Rightarrow$, les constantes propositionnelles $\perp$ et $\top$ et les quantificateurs $\forall$ et $\exists$. On rappelle que la forme normale de négation d'une formule est une formule équivalente qui n'utilise pas le connecteur $\Rightarrow$ et dans laquelle la négation ne porte que sur des symboles de prédicat (ici P et Q).

(a) Mettre sous forme normale de négation les formules suivantes :

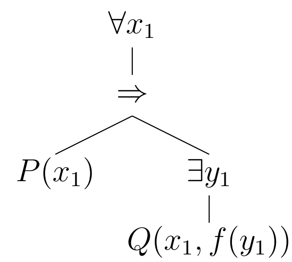
i. $\forall x, (P(x) \Rightarrow \neg \exists y, (Q(x,y) \Rightarrow \neg P(y)))$

ii. $((\forall x, Q(x, f(x))) \Rightarrow \exists y, P(y)) \Rightarrow \forall x, \exists y, Q(x,y)$

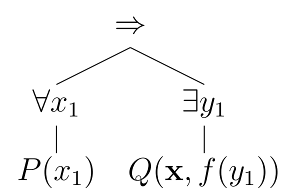
(b) Spécifier par des équations récursives les deux fonctions `fnn` et `fnn-neg` telles que si A est une formule du calcul des prédicats, alors `fnn`(A) est une formule en forme normale de négation logiquement équivalente à A et `fnn-neg`(A) est une formule en forme normale de négation logiquement équivalente à $\neg A$.

Correction :

1. (a) formule correcte (variables liées renommées)



(b) formule correcte (variable libre $\mathbf{x}$ *en gras)*



(c) la formule $\forall x, (P(x) \Rightarrow \exists y, Q(f(x,y)))$ n'est pas bien fomée, il y a deux arguments à la fonction f qui est d'arité 1 et un seul argument au prédicat Q qui est d'arité 2.

2. (a) i.

$$\begin{aligned}\forall x, (P(x) \Rightarrow \neg \exists y, (Q(x,y) \Rightarrow \neg P(y))) &\equiv \forall x, (\neg P(x) \vee \forall y, \neg(Q(x,y) \Rightarrow \neg P(y)))\\ &\equiv \forall x, (\neg P(x) \vee \forall y, (Q(x,y) \wedge P(y)))\end{aligned}$$

ii.

$$\begin{aligned}((\forall x, Q(x, f(x))) \Rightarrow \exists y, P(y)) \Rightarrow \forall x, \exists y, Q(x,y) &\equiv\\ \neg((\forall x, Q(x, f(x))) \Rightarrow \exists y, P(y)) \vee \forall x, \exists y, Q(x,y) &\equiv\\ ((\forall x, Q(x, f(x))) \wedge \neg \exists y, P(y)) \vee \forall x, \exists y, Q(x,y) &\equiv\\ ((\forall x, Q(x, f(x))) \wedge \forall y, \neg P(y)) \vee \forall x, \exists y, Q(x,y) &\end{aligned}$$

(b)

$$\begin{array}{rlrl}
\textit{fnn}(P) & = P & \textit{fnn-neg}(P) & = \neg P \quad \textit{si } P \textit{ atomique}\\
\textit{fnn}(\neg P) & = \textit{fnn-neg}(P) & \textit{fnn-neg}(\neg P) & = \textit{fnn}(P)\\
\textit{fnn}(P \wedge Q) & = \textit{fnn}(P) \wedge \textit{fnn}(Q) & \textit{fnn-neg}(P \wedge Q) & = \textit{fnn-neg}(P) \vee \textit{fnn-neg}(Q)\\
\textit{fnn}(P \vee Q) & = \textit{fnn}(P) \vee \textit{fnn}(Q) & \textit{fnn-neg}(P \vee Q) & = \textit{fnn-neg}(P) \wedge \textit{fnn-neg}(Q)\\
\textit{fnn}(P \Rightarrow Q) & = \textit{fnn-neg}(P) \vee \textit{fnn}(Q) & \textit{fnn-neg}(P \Rightarrow Q) & = \textit{fnn}(P) \wedge \textit{fnn-neg}(Q)\\
\textit{fnn}(\forall x, P) & = \forall x, \textit{fnn}(P) & \textit{fnn-neg}(\forall x, P) & = \exists x, \textit{fnn-neg}(P)\\
\textit{fnn}(\exists x, P) & = \exists x, \textit{fnn}(P) & \textit{fnn-neg}(\exists x, P) & = \forall x, \textit{fnn-neg}(P)
\end{array}$$

Exercice 4 *Modélisation énigme (4 points).* Un fermier doit faire traverser une rivière à une chèvre, un loup et un chou. Il dispose d'une barque qui ne peut contenir que l'un des trois (en plus de lui-même). Par ailleurs il ne peut laisser seuls ensemble sur une rive, ni la chèvre et le chou, ni le loup et la chèvre.

1. On cherche d'abord à modéliser les ensembles d'objets. Pour cela on se donne dans la signature, trois constantes `chèvre`, `chou` et `loup` et une constante `emp` pour l'ensemble vide. On se donne aussi deux opérations sur les ensembles `add` qui ajoute un élément à un ensemble et `rem` qui supprime un élément d'un ensemble.
   (a) Préciser l'arité des différentes opérations.
   (b) Donner un terme qui représente l'ensemble T formé des trois (chèvre, loup et chou) ainsi que l'ensemble L qui ne contient que le loup.
2. On se donne deux prédicats binaires $\in$ et $=$ pour exprimer qu'un élément appartient à un ensemble et que deux objets sont égaux. Donner les formules du calcul des prédicats qui représentent les propriétés des différents objets de la signature :
   (a) il n'y a pas d'élément dans l'ensemble vide ;
   (b) caractérisation des éléments de l'ensemble x auquel on a ajouté l'élément a
   (c) caractérisation des éléments de l'ensemble x auquel on a supprimé l'élément a
   (d) deux ensembles sont égaux si et seulement si ils ont les mêmes éléments
3. On cherche maintenant à modéliser le problème de la traversée. Pour cela on se donne un symbole de relation binaire $\mathtt{state}(x, y)$ qui exprime une situation dans laquelle les éléments de l'ensemble x sont à gauche de la rivière et ceux de y sont à droite de la rivière.

   Le but de l'exercice est de modéliser les propriétés de `state` et de démontrer que :
   $\mathtt{state}(T, \mathtt{emp}) \Rightarrow \mathtt{state}(\mathtt{emp}, T)$
   c'est-à-dire que l'on peut passer d'une situation où tout le monde est à gauche à une situation dans laquelle tout le monde est à droite.
   (a) Définir une formule logique $ok(x)$ qui exprime que l'ensemble x est “stable” dans le sens que si x contient la chèvre alors il ne contient ni le chou ni le loup.
   (b) Combien y-a-t-il d'ensembles formés uniquement d'éléments choisis parmi la chèvre, le chou et le loup ? Parmi ces ensembles, combien satisfont la propriété ok ?
   (c) On peut passer de la situation $\mathtt{state}(x, y)$ à la situation $\mathtt{state}(x', y')$ soit en faisant passer un élément a de gauche à droite soit en le faisant passer de droite à gauche.

      Dans le cas d'une traversée de gauche à droite, on doit avoir que a est dans l'ensemble x, que x' est l'ensemble x moins a et y' est l'ensemble y plus a. Il faut également que l'on ait pu laisser y à droite et x' à gauche sans surveillance et donc que les ensembles y et x' vérifient la propriété ok.

      Dans le cas d'une traversée de droite à gauche on doit avoir que a est dans l'ensemble y, que y' est l'ensemble y moins a et x' est l'ensemble x plus a. Il faut également que les ensembles x et y' vérifient la propriété ok.

      Traduire les deux propriétés précédentes en deux formules G et D de la forme :

$$\forall axy, (a \in \ldots \wedge ok(\ldots) \wedge \ldots \wedge \mathtt{state}(x, y) \Rightarrow \mathtt{state}(\ldots, \ldots))$$

(d) Enumérer toutes les conséquences closes (sans variable) des deux formules précédentes de la forme $\texttt{state}(t,u) \Rightarrow \texttt{state}(t',u')$ dans lesquelles on a de plus que l'union des ensembles représentés par t et u est bien l'ensemble T entier et que ces deux ensembles sont disjoints (on pourra utiliser les notations usuelles $\{\texttt{chou},\texttt{loup}\}$ pour représenter les ensembles).

(e) En supposant que les deux formules G et D sont vérifiées, en déduire par la méthode de votre choix, la vérité de la formule :

$$\texttt{state}(T,\texttt{emp}) \Rightarrow \texttt{state}(\texttt{emp},T)$$

Correction :

1. (a) *add et rem sont d'arité 2*

   (b) $T = \texttt{add}(\texttt{chèvre}, \texttt{add}(\texttt{chou}, \texttt{add}(\texttt{loup}, \texttt{emp})))$, $L = \texttt{add}(\texttt{loup}, \texttt{emp})$

2. (a) $\forall e, \neg e \in \texttt{emp}$

   (b) $\forall axe, (e \in \texttt{add}(a,e) \Leftrightarrow (e = a \vee e \in x))$

   (c) $\forall axe, (e \in \texttt{rem}(a,e) \Leftrightarrow (\neg e = a \wedge e \in x))$

   (d) $\forall xy, (x = y) \Rightarrow (\forall e, (e \in x \Leftrightarrow e \in y))$

3. (a) $ok(x) \stackrel{\text{def}}{=} \texttt{chevre} \in x \Rightarrow (\neg\,\texttt{loup} \in x \wedge \neg\,\texttt{chou} \in x)$

   (b) *Il y a* $2^3 = 8$ *sous-ensembles de* T*. Les 5 ensembles suivants sont stables : l'ensemble vide,* $\emptyset$*, les trois ensembles formés d'un seul élément et l'ensemble à deux éléments* $\{\texttt{loup}, \texttt{chou}\}$ *tous les autres ensembles ne sont pas stables.*

   (c)

$$G \stackrel{\text{def}}{=} \forall axy, (a \in x \wedge ok(y) \wedge ok(\texttt{rem}(a,x)) \wedge \texttt{state}(x,y) \Rightarrow \texttt{state}(\texttt{rem}(a,x), \texttt{add}(a,y))$$

$$D \stackrel{\text{def}}{=} \forall axy, (a \in y \wedge ok(x) \wedge ok(\texttt{rem}(a,y)) \wedge \texttt{state}(x,y) \Rightarrow \texttt{state}(\texttt{add}(a,x), \texttt{rem}(a,y))$$

   (d) *Pour instancier la formule* G *il faut que* $\texttt{ok}(y)$ *soit vérifié ce qui nous laisse au plus 5 possibilités. Comme* x *et* y *doivent être deux ensembles disjoints qui forment à eux deux l'ensemble entier on a comme possibilités :*
   - $y = \emptyset$, $x = T$, *la seule possibilité pour que* $\texttt{ok}(\texttt{rem}(a,x))$ *est que* $a = \texttt{chevre}$ *et cela donne*
     $\texttt{state}(T,\emptyset) \Rightarrow \texttt{state}(\{\texttt{chou}, \texttt{loup}\}, \{\texttt{chevre}\})$
   - $y = \{\texttt{chou}\}$, $x = \{\texttt{chevre}, \texttt{loup}\}$, *on a deux possibilités pour* a *:* $a = \texttt{chevre}$ *et* $a = \texttt{loup}$ *on obtient*
     $\texttt{state}(\{\texttt{chevre}, \texttt{loup}\}, \{\texttt{chou}\}) \Rightarrow \texttt{state}(\{\texttt{loup}\}, \{\texttt{chevre}, \texttt{chou}\})$
     $\texttt{state}(\{\texttt{chevre}, \texttt{loup}\}, \{\texttt{chou}\}) \Rightarrow \texttt{state}(\{\texttt{chevre}\}, \{\texttt{loup}, \texttt{chou}\})$
   - *De même si* $y = \{\texttt{chevre}\}$ *et* $y = \{\texttt{loup}\}$ *on obtient les quatre formules suivantes :*
     $\texttt{state}(\{\texttt{chou}, \texttt{loup}\}, \{\texttt{chevre}\}) \Rightarrow \texttt{state}(\{\texttt{loup}\}, \{\texttt{chou}, \texttt{chevre}\})$
     $\texttt{state}(\{\texttt{chou}, \texttt{loup}\}, \{\texttt{chevre}\}) \Rightarrow \texttt{state}(\{\texttt{chou}\}, \{\texttt{loup}, \texttt{chevre}\})$
     $\texttt{state}(\{\texttt{chevre}, \texttt{chou}\}, \{\texttt{loup}\}) \Rightarrow \texttt{state}(\{\texttt{chevre}\}, \{\texttt{chou}, \texttt{loup}\})$
     $\texttt{state}(\{\texttt{chevre}, \texttt{chou}\}, \{\texttt{loup}\}) \Rightarrow \texttt{state}(\{\texttt{chou}\}, \{\texttt{chevre}, \texttt{loup}\})$
   - *Il reste le cas dans lequel* $y = \{\texttt{chou}, \texttt{loup}\}$*, on a alors* $x = \{\texttt{chevre}\}$ *et* $a = \texttt{chevre}$ *cela donne*
     $\texttt{state}(\{\texttt{chevre}\}, \{\texttt{loup}, \texttt{chou}\}) \Rightarrow \texttt{state}(\emptyset, \{\texttt{chevre}, \texttt{loup}, \texttt{chou}\})$

- *On a une situation symétrique pour les instances de la formule D pour lesquelles on s'intéresse uniquement aux cas où x vérifie la propreté ok.*
  $\mathtt{state}(\emptyset, T) \Rightarrow \mathtt{state}(\{\mathtt{chevre}\}, \{\mathtt{chou}, \mathtt{loup}\})$
  $\mathtt{state}(\{\mathtt{chou}\}, \{\mathtt{chevre}, \mathtt{loup}\}) \Rightarrow \mathtt{state}(\{\mathtt{chevre}, \mathtt{chou}\}, \{\mathtt{loup}\})$
  $\mathtt{state}(\{\mathtt{chou}\}, \{\mathtt{chevre}, \mathtt{loup}\}) \Rightarrow \mathtt{state}(\{\mathtt{loup}, \mathtt{chou}\}, \{\mathtt{chevre}\})$
  $\mathtt{state}(\{\mathtt{chevre}\}, \{\mathtt{chou}, \mathtt{loup}\},) \Rightarrow \mathtt{state}(\{\mathtt{chou}, \mathtt{chevre}\}, \{\mathtt{loup}\})$
  $\mathtt{state}(\{\mathtt{chevre}\}, \{\mathtt{chou}, \mathtt{loup}\}) \Rightarrow \mathtt{state}(\{\mathtt{loup}, \mathtt{chevre}\}, \{\mathtt{chou}\})$
  $\mathtt{state}(\{\mathtt{loup}\}, \{\mathtt{chevre}, \mathtt{chou}\}) \Rightarrow \mathtt{state}(\{\mathtt{chou}, \mathtt{loup}\}, \{\mathtt{chevre}\})$
  $\mathtt{state}(\{\mathtt{loup}\}, \{\mathtt{chevre}, \mathtt{chou}\}) \Rightarrow \mathtt{state}(\{\mathtt{chevre}, \mathtt{loup}\}, \{\mathtt{chou}\})$
  $\mathtt{state}(\{\mathtt{loup}, \mathtt{chou}\}, \{\mathtt{chevre}\}) \Rightarrow \mathtt{state}(\{\mathtt{chevre}, \mathtt{loup}, \mathtt{chou}\}, \emptyset)$

(e) *On utilise des étapes de raisonnement (modus-ponens) en utilisant les instances de G et D précédentes :*
  - *on part de* $\mathtt{state}(T, \emptyset)$
  - *On utilise G avec* $y = \emptyset$: $\mathtt{state}(T, \emptyset) \Rightarrow \mathtt{state}(\{\mathtt{chou}, \mathtt{loup}\}, \{\mathtt{chevre}\})$
    on obtient $\mathtt{state}(\{\mathtt{chou}, \mathtt{loup}\}, \{\mathtt{chevre}\})$
  - *On utilise G avec* $y = \{\mathtt{chevre}\}$ *et* $a = \mathtt{chou}$:
    $\mathtt{state}(\{\mathtt{chou}, \mathtt{loup}\}, \{\mathtt{chevre}\}) \Rightarrow \mathtt{state}(\{\mathtt{loup}\}, \{\mathtt{chou}, \mathtt{chevre}\})$
    on obtient $\mathtt{state}(\{\mathtt{loup}\}, \{\mathtt{chou}, \mathtt{chevre}\})$
    (le fait qu'il y ait deux étapes gauches consécutives correspond à un retour à vide).
  - *On utilise D avec* $x = \{\mathtt{loup}\}$ *et* $a = \mathtt{chevre}$:
    $\mathtt{state}(\{\mathtt{loup}\}, \{\mathtt{chevre}, \mathtt{chou}\}) \Rightarrow \mathtt{state}(\{\mathtt{chevre}, \mathtt{loup}\}, \{\mathtt{chou}\})$
    on obtient $\mathtt{state}(\{\mathtt{chevre}, \mathtt{loup}\}, \{\mathtt{chou}\})$
  - *On utilise G avec* $y = \{\mathtt{chou}\}$ *et* $a = \mathtt{loup}$:
    $\mathtt{state}(\{\mathtt{chevre}, \mathtt{loup}\}, \{\mathtt{chou}\}) \Rightarrow \mathtt{state}(\{\mathtt{chevre}\}, \{\mathtt{loup}, \mathtt{chou}\})$
    on obtient $\mathtt{state}(\{\mathtt{chevre}\}, \{\mathtt{loup}, \mathtt{chou}\})$
  - *On utilise G avec* $y = \{\mathtt{loup}, \mathtt{chou}\}$ *et* $a = \mathtt{chevre}$:
    $\mathtt{state}(\{\mathtt{chevre}\}, \{\mathtt{loup}, \mathtt{chou}\}) \Rightarrow \mathtt{state}(\emptyset, \{\mathtt{chevre}, \mathtt{loup}, \mathtt{chou}\})$
    On obtient la conclusion attendue $\mathtt{state}(\emptyset, T)$

Exercice 5 *Satisfiabilité (4 points).*
Soit la formule $A \stackrel{\text{def}}{=} (\exists x, P(x)) \Rightarrow ((\exists x, Q(x)) \Rightarrow \exists x, (P(x) \wedge Q(x)))$

1. Mettre la formule A en forme normale de négation (la définition de la forme normale de négation est redonnée à l'exercice 3).
2. Soit B la formule obtenue à la question précédente, dire si les affirmations suivantes sont vraies ou fausses (sans les justifier, c'est du cours !)
   - $A \models B$
   - $B \models A$
   - A est satisfiable si et seulement si B l'est.
3. Eliminer dans B les quantificateurs existentiels ($\exists$) en utilisant la skolémisation et mettre les quantificateurs universels ($\forall$) en tête de la formule.
4. Soit C la formule obtenue à la question précédente, dire si les affirmations suivantes sont vraies ou fausses (sans les justifier, c'est du cours !)
   - $B \models C$
   - $C \models B$

– B est satisfiable si et seulement si C l'est.

5. La formule C est-elle vraie dans un modèle dont le domaine a exactement un élément ? Même question pour A, justifier vos réponses.
6. La formule A est-elle satisfiable ?
7. Mettre la formule $\neg A$ en forme clausale.
8. Indiquer le domaine et la base de Herbrand pour la forme clausale ainsi obtenue.
9. La formule $\neg A$ est-elle satisfiable ?
10. La formule A est-elle valide ?

Correction :

1. *$B = (\forall x, \neg P(x)) \vee (\forall x, \neg Q(x)) \vee \exists x, (P(x) \wedge Q(x))$.*
2. *On a $A \equiv B$ donc les trois affirmations sont vérifiées.*
3. *$C = \forall xy, (\neg P(x) \vee \neg Q(y) \vee (P(a) \wedge Q(a)))$*
4. *On a $C \models B$ mais pas $B \models C$ et B satisfiable si et seulement si C l'est.*
5. *Dans une interprétation I à un seul élément e, la formule C est vrai si et seulement si on a $\neg P_I(e) \vee \neg Q_I(e) \vee (P_I(e) \wedge Q_I(e)))$ qui est toujours vrai. Donc C est vraie dans toutes les interprétations à un élément. Comme C est vraie dans ce modèle, il en est de même de B et donc de A.*
6. *La formule A est satisfiable.*
7. *On a $\neg A \equiv (\exists x, P(x)) \wedge (\exists x, Q(x)) \wedge (\forall x, (\neg P(x) \vee \neg Q(x)))$. Après skolémisation, on obtient les trois clauses $C_1 = P(a)$, $C_2 = Q(b)$ et $C_3 = \neg P(x) \vee \neg Q(x)$.*
8. *Le domaine de Herbrand est $D_H = \{a, b\}$ et la base de Herbrand est $B_H = \{P(a), P(b), Q(a), Q(b)\}$.*
9. *On choisit comme interprétation $P_H(a) = Q_H(b) = V$ et $P_H(b) = Q_H(a) = F$ et on vérifie que cette interprétation rend vraies les clauses et donc la formule $\neg A$.*
   La formule $\neg A$ est satisfiable.
10. *La formule A n'est pas valide car sinon $\neg A$ serait insatisfiable.*