Corrigé 1° session
de Programmation logique et Logique

1 Prolog

1.1 Sous-listes

Soit `L = [T|Q]`. Les sous-listes de `L` de longueur `P` sont
- soit des sous-listes `[T|R]` où `R` est une sous-liste de longueur `P-1` de `Q`
- soit des sous-listes de `Q` de longueur `P`

Cette définition se traduit en Prolog par :

```
% sousliste(L,P,M) vrai ssi M sous-liste de L de longueur P
sousliste([],0,[]).
sousliste([T|Q],P,[T|R]):-sousliste(Q,P-1,R).
sousliste([T|Q],P,R):-sousliste(Q,P,R).
```

On donne un exemple de fonctionnement du programme :

```
>> sousliste([3],1,R).
R = [3].
```

1.2 Analyse syntaxique

1.2.1 reconnaisseur

```
rec_S(X):-rec_S(X,[]).
rec_S(X0,X1):-a(X0,X1).
rec_S(X0,X3):-b(X0,X1),rec_S(X1,X2),rec_S(X2,X3).
a([a|X],X).
b([b|X],X).
```

1.2.2 trace

L'arbre d'exécution est représentée avec des tabulations. Les noms `r1` et `r2` désignent les 2 règles de `rec_S/2`. Trace de `rec_S([b,a,a])`.

```
rec_S([b,a,a]).
rec_S([b,a,a],[]).
  r1 : a([b,a,a],[]).
  échec
  r2 : b([b,a,a],X1),rec_S(X1,X2),rec_S(X2,[]).
       rec_S([a,a],X2),rec_S(X2,[]).
       r1 : a([a,a],X2),rec_S(X2,[]).
            rec_S([a],[]).
            r1 : a([a],[]).
            réussite

            r2 : b([a],X1),rec_S(X1,X2),rec_S(X2,[]).
            échec
       r2 : b([a,a],X1'),rec_S(X1',X2'),rec_S(X2',X2),rec_S(X2,[])
            Renommage necessaire
            échec
```

1.2.3 traducteur

```
trad_S(X,A):-trad_S(X,[],A).
trad_S(X0,X1,s(a)):-a(X0,X1).
trad_S(X0,X3,s(b,A1,A2)):-b(X0,X1),trad_S(X1,X2,A1),trad_S(X2,X3,A2).
```

1.3 Contraintes

1.3.1 diag1elt

```
% diag1elt(X1,Y1,[X2,...Xk],Y2) vrai ssi la reine X1,Y1 n'est pas sur la
% même diagonale gauche-droite que (X2,Y2) .. (X3,Y2+1),...(Xk,Y2+k-1)
diag1elt(X,Y,[],Z).
diag1elt(X1,Y1,[X2|Q],Y2):-
        dif(X1+Y1,X2+Y2), diag1elt(X1,Y1,Q,Y2+1).
```

1.3.2 diag1

```
% diag1([X1,...Xk],Y) vrai ssi les reines aux positions
% (X1,Y),(X2,Y+1) ...(Xk,Y+k-1) sont sur des diagonales gauche-droite distinctes
diag1([T],Y).
diag1([X1,X2|Q],Y):- diag1elt(X1,Y, [X2|Q],Y+1),diag1([X2|Q],Y+1).
```

1.3.3 reines

On donne ci-dessous le prédicat `reines` ainsi que les prédicats qu'il utilise. On a rappelé la spécification (qui était demandée) et la définition (qui ne l'était pas) de tous prédicats auxiliaires.

```
% L est une liste des indices des lignes pour les colonnes de
% 1 à size(L)
reines(L):-
        tousdiff(L),    %les reines sont sur des lignes distinctes
        diag1(L,1),     %les reines sont sur des diagonales gauche-droite distinctes
        diag2(L,1),     %les reines sont sur des diagonales droite-gauche distinctes
        entre(L,1,size(L)),
        enumliste(L).

% diag1([X1,...Xk],Y) vrai ssi les reines aux positions
% (X1,Y),(X2,Y+1) ...(Xk,Y+k-1) sont sur des diagonales gauche-droite distinctes
diag1([T],Y).
diag1([X1,X2|Q],Y):- diag1elt(X1,Y, [X2|Q],Y+1),diag1([X2|Q],Y+1).
diag1elt(X,Y,[],Z).
diag1elt(X1,Y1,[X2|Q],Y2):-
        dif(X1+Y1,X2+Y2),diag1elt(X1,Y1,Q,Y2+1).

% diag2 analogue à diag1 pour les diagonales droite-gauche
diag2([T],Y).
diag2([X1,X2|Q],Y):- diag2elt(X1,Y, [X2|Q],Y+1),diag2([X2|Q],Y+1).
diag2elt(X,Y,[],Z).
diag2elt(X1,Y1,[X2|Q],Y2):-
        dif(X1-Y1,X2-Y2),diag1elt(X1,Y1,Q,Y2+1).

% entre(L,BI,BS) impose que tous les éléments de la liste L sont entre BI et BS
entre([],BI,BS).
entre([T|Q],BI,BS):- lelin(BI,T),lelin(T,BS), entre(Q,BI,BS).

% nonelt(X,L) impose que X est différent de tous les éléments de la liste L
nonelt(X,[]).
nonelt(X,[T|Q]):-dif(X,T),nonelt(X,Q).

% tousdiff(L) impose que tous les éléments de la liste L sont distincts
tousdiff([T]).
tousdiff([T1,T2|Q]):-nonelt(T1,[T2|Q]),tousdiff([T2|Q]).

% enumliste(L) provoque l'énumeration de tous les éléments de la liste L
enumliste([]).
enumliste([T|Q]):-enumlin(T),enumliste(Q).
```

2 Logique

2.1 Validité en logique propositionnelle

2.1.1 Table de vérité

On constate que sur toutes les lignes de la table de vérité, la formule vaut 1. Donc cette formule est valide.

2.1.2 Première transformation en formule normale conjonctive

$(a \lor b) \Rightarrow ((\neg a \land \neg b) \Rightarrow c)$
Par élimination des implications, on obtient : $\equiv \neg(a \lor b) \lor \neg(\neg a \land \neg b) \lor c$
Par déplacement des négations, on obtient : $\equiv (\neg a \land \neg b) \lor a \lor b \lor c$
Par distributivité de $\land$ sur $\lor$, on obtient : $\equiv (\neg a \lor a \lor b \lor c) \land (\neg b \lor a \lor b \lor c)$
Par suppression des disjonctions valides, on obtient : $\equiv \top$
Donc la formule initiale est équivalente à $\top$, et par suite elle est valide.

2.1.3 Deuxième transformation en formule normale conjonctive

$(a \lor b) \Rightarrow ((\neg a \lor \neg b) \Rightarrow c)$
Par élimination des implications, on obtient : $\equiv \neg(a \lor b) \lor \neg(\neg a \lor \neg b) \lor c$
Par déplacement des négations, on obtient : $\equiv (\neg a \land \neg b) \lor (a \land b) \lor c$
Par distributivité de $\land$ sur $\lor$, on obtient : $((\neg a \lor a) \land (\neg a \lor b) \land (\neg b \lor a) \land (\neg b \lor b)) \lor c$
Par suppression des disjonctions valides, on obtient : $((\neg a \lor b) \land (\neg b \lor a)) \lor c$
Par distributivité de $\land$ sur $\lor$, on obtient : $(\neg a \lor b \lor c) \land (\neg b \lor a \lor c)$
Cette formule a comme contre-modèle $a = 1, b = 0, c = 0$, elle n'est donc pas valide.

2.1.4 Première application de la méthode des tableaux

<table>
<tr><td colspan="4">(1) $\neg(((a \Rightarrow b) \vee (c \Rightarrow d)) \Rightarrow ((a \vee c) \Rightarrow (b \vee d)))$*
(2) $(a \Rightarrow b) \vee (c \Rightarrow d)$* de (1)
(3) $\neg((a \vee c) \Rightarrow (b \vee d))$* de (1)
(4) $a \vee c$* de (3)
(5) $\neg(b \vee d)$* de (3)
(6) $\neg b$* de (5)
(7) $\neg d$* de (5)</td></tr>
<tr><td colspan="3">(8) $a \Rightarrow b$* de (2)
(4) $a \vee c$*
(6) $\neg b$*
(7) $\neg d$*</td><td rowspan="3">(8) $c \Rightarrow d$ de (2)
(4) $a \vee c$
(6) $\neg b$
(7) $\neg d$</td></tr>
<tr><td colspan="2">(9)$\neg a$ de (8)
(4)$a \vee c$*
(8)$\neg b$
(7)$\neg d$</td><td rowspan="2">(9) b de (8)
(4)$a \vee c$
(8) $\neg b$
(7) $\neg d$</td></tr>
<tr><td>(10) a de (4)
(9) $\neg a$
(8) $\neg b$
(7) $\neg d$
X</td><td>(10) c de (4)
(9) $\neg a$
(8) $\neg b$
(7) $\neg d$</td></tr>
</table>

La première branche ouverte donne un contre modèle $c, \neg a, \neg b, \neg d$ de la formule, c'est-à-dire, $c = 1, a = 0, b = 0, d = 0$. Il est alors inutile de continuer la construction.

En évitant les copies de formules (qui sont alors partagées entre les branches), on obtient un développement plus court de la preuve (voir ci-dessous). Chaque fois qu'une règle est appliquée à une formule, les résultats de cette application sont copiés dans toutes les branches où figure la formule. Par exemple l'application d'une règle à la formule (4) a été effectuée dans les 3 branches existantes au moment de cette application.

<table>
<tr><td colspan="6">(1) $\neg(((a \Rightarrow b) \vee (c \Rightarrow d)) \Rightarrow ((a \vee c) \Rightarrow (b \vee d)))$*
(2) $(a \Rightarrow b) \vee (c \Rightarrow d)$* de (1)
(3) $\neg((a \vee c) \Rightarrow (b \vee d))$* de (1)
(4) $a \vee c$* de (3)
(5) $\neg(b \vee d)$* de (3)
(6) $\neg b$ de (5)
(7) $\neg d$ de (5)</td></tr>
<tr><td colspan="4">(8) $a \Rightarrow b$* de (2)</td><td colspan="2">(8) $c \Rightarrow d$ de (2)</td></tr>
<tr><td colspan="2">(9)$\neg a$ de (8)</td><td colspan="2">(9) b de (8)</td><td rowspan="2">(9) a de (4)</td><td rowspan="2">(9) c de (4)</td></tr>
<tr><td>(10) a de (4)
X</td><td>(10) c de (4)</td><td>(10) a de (4)</td><td>(10) c de (4)</td></tr>
</table>

Dans la suite, on est fidèle à la disposition donnée en annexe de l'examen, même si elle implique beaucoup de copies de formules.

2.1.5 Deuxième application de la méthode des tableaux

<table>
<tr><td colspan="2">(1) $\neg(((a \vee c) \Rightarrow (b \vee d)) \Rightarrow ((a \Rightarrow b) \vee (c \Rightarrow d)))^*$
(2) $(a \vee c) \Rightarrow (b \vee d)^*$ de (1)
(3) $\neg((a \Rightarrow b) \vee (c \Rightarrow d))^*$ de (1)
(4) $\neg(a \Rightarrow b)^*$ de (3)
(5) $\neg(c \Rightarrow d)^*$ de (3)
(6) a^* de (4)
(7) $\neg b^*$ de (4)
(8) c^* de (5)
(9) $\neg d^*$ de (5)</td><td></td></tr>
<tr><td rowspan="2">(10) $\neg(a \vee c)^*$ de (2)
(6) a
(7) $\neg b$
(8) c
(9) $\neg d$
(11)$\neg a$ de (10)
(12)$\neg c$ de (10)
X</td><td colspan="2">(10) $b \vee d^*$ de (2)
(6) a^*
(7) $\neg b^*$
(8) c^*
(9) $\neg d$</td></tr>
<tr><td>(11) b de (10)
(6) a
(7) $\neg b$
(8) c
(9) $\neg d$
X</td><td>(11) d de (10)
(6) a
(7) $\neg b$
(8) c
(9) $\neg d$
X</td></tr>
</table>

Toutes les branches du tableau sont fermées, donc la formule est valide.

2.2 Évaluation

Dans la suite, pour $i, j \in \{1, 2, 3\}$, on identifie $R(i, j)$ et la valeur de vérité de $(i, j) \in R^I$ ainsi que les symboles logiques avec leur sens.

Il est utile de dessiner la relation, sous forme de flèches $1 \rightarrow 2 \longleftrightarrow 3$. On voit alors (sans calcul) que

- la formule (1) vaut 1 (de tout élément du domaine part une flèche)
- la formule (2) vaut 0 (car il y a un élément du domaine où n'arrive pas de flèche)
- la formule (3) vaut 1 (car la relation est une fonction)
- la formule (4) vaut 0 (car cette fonction n'est pas injective)

Formule $\forall x \exists y R(x, y)$:
(a) $\forall x \exists y R(x, y) = (\exists y R(1, y)) \wedge (\exists y R(2, y)) \wedge (\exists y R(2, y))$
(b) $\exists y R(1, y) = R(1, 1) \vee R(1, 2) \vee R(1, 3) = 1$
(c) $\exists y R(2, y) = R(2, 1) \vee R(2, 2) \vee R(2, 3) = 1$
(d) $\exists y R(3, y) = R(3, 1) \vee R(3, 2) \vee R(3, 3) = 1$.
D'après a, b, c, d, la formule vaut 1 dans l'interprétation I.

Formule $\forall x \exists y R(y, x)$:
(a) $\forall x \exists y R(y, x) = (\exists y R(y, 1)) \wedge (\exists y R(y, 2)) \wedge (\exists y R(y, 3))$
(b) $\exists y R(y, 1) = R(1, 1) \vee R(2, 1) \vee R(3, 1) = 0$
D'après a, b la formule vaut 0 dans l'interprétation I.

Formule $\forall x \forall y \forall z((R(x,y) \wedge R(x,z)) \Rightarrow (y = z))$.
La formule $(R(x,y) \wedge R(x,z))$ n'est vraie que dans les 3 cas suivants :
- $x = 1, y = 2, z = 2$
- $x = 2, y = 3, z = 3$
- $x = 3, y = 2, z = 2$

Pour toutes ces assignations $y = z$ est vraie.
Par suite $((R(x,y) \wedge R(x,z)) \Rightarrow (y = z))$ est vraie dans toute assignation de ses variables, donc $\forall x \forall y \forall z((R(x,y) \wedge R(x,z)) \Rightarrow (y = z))$ vaut 1 dans l'interprétation I.

Formule $\forall x \forall y \forall z((R(x,z) \wedge R(y,z)) \Rightarrow (x = y))$.
La formule $((R(x,z) \wedge R(y,z)) \Rightarrow (x = y))$ vaut 0 pour $x = 1, y = 2, z = 3$.
Donc la formule $\forall x \forall y \forall z((R(x,z) \wedge R(y,z)) \Rightarrow (x = y))$ vaut 0 dans l'interprétation I.

2.3 Modèle

Considérons l'interprétation ayant pour domaine l'ensemble de réels et donnant à R le sens de la relation $<$ sur les réels. Cette intreprétation est un modèle des 4 formules, car il y a au moins 2 réels dont l'un est inférieur à l'autre (formule 1), la relation $<$ est transitive (formule 2), entre 2 réels distincts, on peut placer un autre réel (formule 3), la relation $<$ est irréflexive (formule 4).

2.4 Validité en logique du premier ordre

En fait les deux formules sont valides, et on le prouve par la méthode des tableaux. Dans cette méthode, les lettres a et b sont des constantes.

(1) $\neg(((\exists x P(x)) \Rightarrow (\exists x Q(x))) \Rightarrow (\exists x(P(x) \Rightarrow Q(x))))^*$ (2) $(\exists x P(x)) \Rightarrow (\exists x Q(x))^*$ de (1) (3) $\neg(\exists x(P(x) \Rightarrow Q(x)))^*$ de (1)	
(4) $\neg(\exists x P(x))$ de (2)	(4) $\exists x Q(x)^*$ de (2)
(3) $\neg(\exists x(P(x) \Rightarrow Q(x)))$	(3) $\neg(\exists x(P(x) \Rightarrow Q(x)))$
(5) $\neg P(a)$ de (4)	(5) $Q(a)$ de (4)
(6) $\neg(P(a) \Rightarrow Q(a))$ de (3)	$\neg(P(a) \Rightarrow Q(a))$ de (3)
(7) $P(a)$ de (6)	(7) $P(a)$ de (6)
(8) $\neg Q(a)$ de (6)	(8)$\neg Q(a)$ de (6)
X	X

<table>
<tr><td colspan="2">(1) $\neg(((\exists x P(x)) \Rightarrow (\exists x Q(x))) \Rightarrow (\exists x \forall y (P(y) \Rightarrow Q(x))))^*$
(2) $(\exists x P(x)) \Rightarrow (\exists x Q(x))^*$ de (1)
(3) $\neg(\exists x \forall y (P(y) \Rightarrow Q(x)))^*$ de (1)</td></tr>
<tr><td>(4) $\neg(\exists x P(x))$ de (2)
(3) $\neg(\exists x \forall y (P(y) \Rightarrow Q(x)))$
(5) $\neg(\forall y (P(y) \Rightarrow Q(a)))^*$ de (3)
(6) $\neg(P(b) \Rightarrow Q(a))^*$ de (5)
(7) $P(b)$ de (7)
(8) $\neg Q(a)$ de (6)
(9) $\neg P(b)$ de (4)
X</td><td>(4) $\exists x Q(x)^*$ de (2)
(3) $\neg(\exists x \forall y (P(y) \Rightarrow Q(x)))$
(5) $Q(a)$ de (4)
(6) $\neg(\forall y (P(y) \Rightarrow Q(a)))^*$ de (3)
(7) $\neg(P(b) \Rightarrow Q(a))^*$ de (5)
(8) $P(b)$ de (7)
(9) $\neg Q(a)$ de (6)
X</td></tr>
</table>