Modélisation et Programmation – Sessions 1 et 3 – 5/11/2019

Prénom : Nom : Groupe :

**Les exercices sont indépendants les uns des autres. Lisez le sujet avant de commencer. Traitez les exercices dans votre ordre de préférence.**
Dans tous les exercices, vous veillerez à justifier vos résultats, par exemple en faisant apparaître le nom des règles utilisées, il en sera tenu compte dans la notation.
Les preuves demandées ont toutes été réalisées avec Coq et Why3 pour réduire les risques d'erreurs. Si vous pensez avoir trouvé une erreur, indiquez le sur votre copie, corrigez la et poursuivez votre composition.

**Exercice 1** Modéliser en logique des prédicats, sous la forme de formules bien formées, l'énoncé suivant : « Les carnivores mangent de la viande. Les requins sont des carnivores. Jacques est un requin. Les végétariens mangent des végétaux. Les omnivores sont à la fois carnivores et végétariens. Donc les omnivores mangent de la viande et des végétaux. »

**Exercice 2** Soient A, B et C des variables propositionnelles, montrer, **en utilisant les propriétés des opérateurs de logique des propositions** rappelées en annexe A, que la formule suivante est une tautologie :

$$(A \wedge B \rightarrow C) \rightarrow (A \rightarrow B \rightarrow C)$$

**Exercice 3** Soient A et B des variables propositionnelles, montrer, **en utilisant la déduction naturelle constructive (c'est-à-dire sans les règles du tiers-exclu (TE dans le résumé de cours) et de l'absurde classique (Abs dans le résumé de cours))**, que la formule suivante est une tautologie :

$$(A \to B \to C) \to (A \land B \to C)$$

**Exercice 4** Soient A et B des variables propositionnelles, montrer, **en utilisant la déduction naturelle classique (y compris les règles du tiers-exclu ou de l'absurde classique)**, que la formule suivante est une tautologie :

$$\neg A \lor B \to \neg A \lor A \land B$$

**Exercice 5** Prouver la correction totale du triplet de Hoare suivant pour un programme calculant la somme des $\mathtt{N}$ premiers entiers naturels. Nous vous suggérons d'exploiter $\mathtt{R} = \dfrac{\mathtt{I} \times (\mathtt{I} + 1)}{2}$ comme invariant et $\mathtt{N} - \mathtt{I}$ comme variant. Vous compléterez l'invariant si nécessaire pour construire la preuve. Vous détaillerez sur la page suivante la preuve des obligations obtenues.

$\{\mathtt{N} \geq 0\}$

```
R := 0;

I := 0

while (I < N) do

    I := I + 1;

    R := R + I

od;
```

$\{\mathtt{R} = \dfrac{\mathtt{N} \times (\mathtt{N} + 1)}{2}\}$

**Exercice 6** Nous considérons la spécification des listes munie des équations étudiées en cours et travaux dirigés :

(a) $\forall l \in \texttt{liste}(A).\ \texttt{append}(\texttt{Nil},l) = l$
(b) $\forall t \in A. \forall l, q \in \texttt{liste}(A).\ \texttt{append}(\texttt{Cons}(t,\ q),l) = \texttt{Cons}(t,\ \texttt{append}(q,l))$
(c) $\forall l \in \texttt{liste}(A).\ \texttt{append}(l,\texttt{Nil}) = l$
(d) $\forall l_1, l_2, l_3 \in \texttt{liste}(A).\ \texttt{append}(l_1,\texttt{append}(l_2,l_3)) = \texttt{append}(\texttt{append}(l_1,l_2),l_3)$

Nous complétons cette spécification par celles des fonctions qui assurent respectivement la séparation des éléments d'une liste et l'applatissement d'une liste de listes.
Le comportement de ces fonction peut être modélisé par les équations suivantes :

(e) $\texttt{flatten}(\texttt{Nil}) = \texttt{Nil}$
(f) $\forall t \in \texttt{liste}(A). \forall q \in \texttt{liste}(\texttt{liste}(A)).\ \texttt{flatten}(\texttt{Cons}(t, q)) = \texttt{append}(t, \texttt{flatten}(q))$
(g) $\texttt{split}(\texttt{Nil}) = \texttt{Nil}$
(h) $\forall t \in A. \forall q \in \texttt{liste}(A).\ \texttt{split}(\texttt{Cons}(t, q)) = \texttt{Cons}(\texttt{Cons}(t, \texttt{Nil}), \texttt{split}(q))$

En utilisant ces équations comme des axiomes et en introduisant si nécessaire des lemmes intermédiaires, montrer que ces opérations vérifient la propriété suivante :

(i) $\forall l \in \texttt{liste}(A).\ \texttt{flatten}(\texttt{split}(l)) = l$

**Exercice 7** Un nombre rationnel est composé d'un numérateur éventuellement suivie d'un dénominateur.
Le numérateur est un nombre entier relatif qui ne doit pas être nul quand il est suivi par un dénominateur. Il débute par un signe optionnel, `+` (plus) ou `-` (moins), suivi par une suite de chiffres non vide.
Un numérateur débute par l'opérateur `/` (divise), suivie par un chiffre différent de zéro et enfin par une suite de chiffres éventuellement vide.

1. Proposer une expression régulière qui modélise le langage des nombres rationnels.

2. Est il possible de modéliser par une expression régulière ou une grammaire les nombres rationnels normalisés (le numérateur et le dénominateur doivent être premiers entre eux) ?

**Exercice 8** Soit la grammaire qui modélise la partie appel de fonctions des expressions en langage C, C++, C#, Java, ..., $G_0 = (A, V, T, P)$ composée des non-terminaux $V = \{L\ E\ T\}$, de l'axiome $E$, des terminaux $A = \{$`entier` `ident` , ( )$\}$ et de l'ensemble $P$ de règles suivantes :

1. $E \rightarrow$ `ident( )`
2. $E \rightarrow$ `ident(` $L$ `)`
3. $E \rightarrow$ `entier`
4. $E \rightarrow$ `ident`
5. $L \rightarrow L$ , $E$
6. $L \rightarrow E$

Voici un exemple de mots modélisés par cette grammaire :

```
f( 1, g( i ), h() }
```

1. Construire la dérivation la plus à gauche de cet exemple selon cette grammaire ;

2. Donner l'arbre de dérivation de cet exemple selon cette grammaire ;

3. Peut on modéliser ce langage par une expression régulière ? Pourquoi ?

4. Donner une grammaire au format graphique de Conway modélisant ce langage.

5. Donner une grammaire au format EBNF modélisant ce langage.

# A Logique des propositions : Vision sémantique

## A.1 Tables de vérité

La sémantique de $\top$ (respectivement $\bot$) est représenté par $V$ (respectivement $F$).

| $A$ | $B$ | $\neg A$ | $A \land B$ | $A \lor B$ | $A \to B$ | $A \leftrightarrow B$ |
|---|---|---|---|---|---|---|
| $F$ | $F$ | $V$ | $F$ | $F$ | $V$ | $V$ |
| $F$ | $V$ | $V$ | $F$ | $V$ | $V$ | $F$ |
| $V$ | $F$ | $F$ | $F$ | $V$ | $F$ | $F$ |
| $V$ | $V$ | $F$ | $V$ | $V$ | $V$ | $V$ |

## A.2 Relation d'équivalence sémantique des formules bien formées

$\varphi = \psi$ si et seulement si les deux formules bien formées $\varphi$ et $\psi$ ont la même table de vérité.

| | |
|---|---|
| | $A \to B = \neg A \lor B$<br>$A \leftrightarrow B = (A \to B) \ \land \ (B \to A)$ |
| Idempotence | $A \land A = A$<br>$A \lor A = A$ |
| | $A \land \neg A = \bot$<br>$A \lor \neg A = \top$<br>$A \land \bot = \bot$<br>$A \land \top = A$<br>$A \lor \bot = A$<br>$A \lor \top = \top$<br>$\neg\neg A = A$ |
| Commutativité | $A \land B = B \land A$<br>$A \lor B = B \lor A$ |
| Associativité | $(A \land B) \land C = A \land (B \land C)$<br>$(A \lor B) \lor C = A \lor (B \lor C)$ |
| Distributivité | $A \land (B \lor C) = (A \land B) \lor (A \land C)$<br>$A \lor (B \land C) = (A \lor B) \land (A \lor C)$ |
| De Morgan | $\neg(A \land B) = \neg A \lor \neg B$<br>$\neg(A \lor B) = \neg A \land \neg B$ |
| Simplification | $A \lor (\neg A \land B) = A \lor B$<br>$A \land (\neg A \lor B) = A \land B$<br>$A \lor (A \land B) = A$<br>$A \land (A \lor B) = A$ |

# B Logique des propositions : Vision syntaxique

## B.1 Déduction naturelle constructive

| Hypothèse | $\dfrac{}{\Gamma,\, \varphi \vdash \varphi}\ \mathit{Hyp}(\Gamma;\varphi)$ |
|---|---|

| Opérateur | Introduction | Elimination |
|---|---|---|
| $\to$ | $\dfrac{\Gamma,\, \varphi \vdash \psi}{\Gamma \vdash \varphi \to \psi}\ I_{\to}(\Gamma;\varphi;\psi)$ | $\dfrac{\Gamma \vdash \varphi \to \psi \quad \Gamma \vdash \varphi}{\Gamma \vdash \psi}\ E_{\to}(\Gamma;\varphi;\psi)$ |
| $\wedge$ | $\dfrac{\Gamma \vdash \varphi \quad \Gamma \vdash \psi}{\Gamma \vdash \varphi \wedge \psi}\ I_{\wedge}(\Gamma;\varphi;\psi)$ | $\dfrac{\Gamma \vdash \varphi \wedge \psi}{\Gamma \vdash \varphi}\ E_{\wedge}^{G}(\Gamma;\varphi;\psi)$ $\dfrac{\Gamma \vdash \varphi \wedge \psi}{\Gamma \vdash \psi}\ E_{\wedge}^{D}(\Gamma;\varphi;\psi)$ |
| $\vee$ | $\dfrac{\Gamma \vdash \varphi}{\Gamma \vdash \varphi \vee \psi}\ I_{\vee}^{G}(\Gamma;\varphi;\psi)$ $\dfrac{\Gamma \vdash \psi}{\Gamma \vdash \varphi \vee \psi}\ I_{\vee}^{D}(\Gamma;\varphi;\psi)$ | $\dfrac{\Gamma \vdash \varphi \vee \psi \quad \Gamma,\, \varphi \vdash \chi \quad \Gamma,\, \psi \vdash \chi}{\Gamma \vdash \chi}\ E_{\vee}(\Gamma;\varphi;\psi;\chi)$ |
| $\neg$ | $\dfrac{\Gamma,\, \varphi \vdash \bot}{\Gamma \vdash \neg\varphi}\ I_{\neg}(\Gamma;\varphi)$ | $\dfrac{\Gamma \vdash \varphi \quad \Gamma \vdash \neg\varphi}{\Gamma \vdash \bot}\ E_{\neg}(\Gamma;\varphi)$ |
| $\bot$ | $\dfrac{\Gamma \vdash \varphi \quad \Gamma \vdash \neg\varphi}{\Gamma \vdash \bot}\ I_{\bot}(\Gamma;\varphi)$ | $\dfrac{\Gamma \vdash \bot}{\Gamma \vdash \varphi}\ E_{\bot}(\Gamma;\varphi)$ |

## B.2 Déduction naturelle classique

| Tiers-exclu | Preuve par l'absurde |
|---|---|
| $\dfrac{}{\Gamma \vdash \varphi \vee \neg\varphi}\ TE(\Gamma;\varphi)$ | $\dfrac{\Gamma,\, \neg\varphi \vdash \bot}{\Gamma \vdash \varphi}\ \mathit{Abs}(\Gamma,\varphi)$ |

# C Logique des prédicats : Vision sémantique

## C.1 Relation d'équivalence sémantique des formules bien formées

$\varphi = \psi$ si et seulement si les deux formules bien formées $\varphi$ et $\psi$ ont la même sémantique, c'est à dire sont valides sémantiquement pour les mêmes modèles $(\forall \mathcal{M}, (\mathcal{M} \models \varphi) \leftrightarrow (\mathcal{M} \models \psi))$.

| | | |
|---|---|---|
| $\mathcal{U} \neq \emptyset$ | $\forall x.\ \varphi = \bigwedge_{x \in \mathcal{U}} \varphi$ | $\exists x.\ \varphi = \bigvee_{x \in \mathcal{U}} \varphi$ |
| $\mathcal{U} = \emptyset$ | $\forall x.\ \varphi = \top$ | $\exists x.\ \varphi = \bot$ |

$$
\begin{array}{rl}
\forall x.\ \varphi = \varphi & x \notin VL(\varphi) \wedge \mathcal{U} \neq \emptyset \\
\exists x.\ \varphi = \varphi & x \notin VL(\varphi) \wedge \mathcal{U} \neq \emptyset \\
\forall x.\ \varphi = \forall y.\ [y/x]\,\varphi & y \text{ inutilisée} \\
\exists x.\ \varphi = \exists y.\ [y/x]\,\varphi & y \text{ inutilisée} \\
\forall x.\ (\forall y.\ \varphi) = \forall y.\ (\forall x.\ \varphi) & \\
\exists x.\ (\exists y.\ \varphi) = \exists y.\ (\exists x.\ \varphi) & \\
\neg(\forall x.\ \varphi) = \exists x.\ (\neg\varphi) & \\
\neg(\exists x.\ \varphi) = \forall x.\ (\neg\varphi) & \\
\forall x.\ (\varphi \to \psi) = (\exists x.\ \varphi) \to \psi & x \notin VL(\psi) \\
\forall x.\ (\varphi \to \psi) = \varphi \to (\forall x.\ \psi) & x \notin VL(\varphi) \\
\exists x.\ (\varphi \to \psi) = (\forall x.\ \varphi) \to \psi & x \notin VL(\psi) \wedge \mathcal{U} \neq \emptyset \\
\exists x.\ (\varphi \to \psi) = \varphi \to (\exists x.\ \psi) & x \notin VL(\varphi) \wedge \mathcal{U} \neq \emptyset \\
\forall x.\ (\varphi \wedge \psi) = (\forall x.\ \varphi) \wedge (\forall x.\ \psi) & \\
\forall x.\ (\varphi \vee \psi) = (\forall x.\ \varphi) \vee \psi & x \notin VL(\psi) \\
\exists x.\ (\varphi \vee \psi) = (\exists x.\ \varphi) \vee (\exists x.\ \psi) & \\
\exists x.\ (\varphi \wedge \psi) = (\exists x.\ \varphi) \wedge \psi & x \notin VL(\psi)
\end{array}
$$

# D Logique des prédicats : Vision syntaxique

## D.1 Déduction naturelle constructive

| Opérateur | Introduction | Elimination |
|---|---|---|
| $\forall.$ | $\dfrac{\Gamma \vdash \varphi}{\Gamma \vdash \forall x.\ \varphi}\ I_{\forall}(\Gamma; x; \varphi)\ (x \notin VL(\Gamma))$ | $\dfrac{\Gamma \vdash \forall x.\ \varphi}{\Gamma \vdash [t/x]\,\varphi}\ E_{\forall}(\Gamma; x; \varphi; t)$ |
| $\exists.$ | $\dfrac{\Gamma \vdash [t/x]\,\varphi}{\Gamma \vdash \exists x.\ \varphi}\ I_{\exists}(\Gamma; x; \varphi)$ | $\dfrac{\Gamma \vdash \exists x.\ \varphi}{\Gamma \vdash [f(\vec{x})/x]\,\varphi}\ E_{\exists}^{SK}(\Gamma; x; \varphi)\ (\vec{x} = VL(\Gamma) \cup VL(\exists x.\ \varphi))$ <br> $\dfrac{\Gamma \vdash \exists x.\ \varphi \qquad \Gamma,\ \varphi \vdash \psi}{\Gamma \vdash \psi}\ E_{\exists}^{MP}(\Gamma; x; \varphi)\ (x \notin VL(\Gamma) \cup VL(\psi))$ |

# E Logique de Floyd/Hoare

$$\frac{}{\{\psi\}\ \text{skip}\ \{\psi\}}\ \text{skip} \qquad\qquad \frac{}{\{[E/x]\,\psi\}\ x := E\ \{\psi\}}\ \text{assign}$$

$$\frac{\{\varphi\}\ P\ \{\chi\} \qquad \{\chi\}\ Q\ \{\psi\}}{\{\varphi\}\ P\ ;Q\ \{\psi\}}\ \text{sequence}$$

$$\frac{\{\varphi \wedge C\}\ P\ \{\psi\} \qquad \{\varphi \wedge \neg C\}\ Q\ \{\psi\}}{\{\varphi\}\ \text{if } C \text{ then } P \text{ else } Q \text{ fi}\ \{\psi\}}\ \text{conditional}$$

$$\frac{\{\varphi \wedge C\}\ P\ \{\varphi\}}{\{\varphi\}\ \text{while } C \text{ invariant } \varphi \text{ do } P \text{ od}\ \{\varphi \wedge \neg C\}}\ \text{partial loop}$$

$$\frac{\{\varphi \wedge C \wedge E \in \mathbb{N} \wedge V = E\}\ P\ \{\varphi \wedge E \in \mathbb{N} \wedge V > E\}}{\{\varphi \wedge E \in \mathbb{N}\}\ \text{while } C \text{ invariant } \varphi \text{ variant } E \text{ do } P \text{ od}\ \{\varphi \wedge \neg C\}}\ \text{total loop}$$

$$\frac{\varphi \rightarrow \chi \qquad \{\chi\}\ P\ \{\psi\}}{\{\varphi\}\ P\ \{\psi\}}\ \text{weaken} \qquad\qquad \frac{\{\varphi\}\ P\ \{\chi\} \qquad \chi \rightarrow \psi}{\{\varphi\}\ P\ \{\psi\}}\ \text{strengthen}$$

# F Expressions régulières : Equivalence sémantique

L'opérateur de concaténation/juxtaposition . est implicite pour ne pas surcharger l'écriture des expressions : $e_1.e_2$ est notée $e_1\,e_2$.

$$\begin{array}{ll}
\emptyset\, e = e\, \emptyset = \emptyset & \Lambda\, e = e\, \Lambda = e \\
e \mid \emptyset = \emptyset \mid e = e & e \mid e = e \\
e_1\,(e_2\,e_3) = (e_1\,e_2)\,e_3 & e_1 \mid (e_2 \mid e_3) = (e_1 \mid e_2) \mid e_3 \\
e_1\,(e_2 \mid e_3) = (e_1\,e_2) \mid (e_1\,e_3) & (e_1 \mid e_2)\,e_3 = (e_1\,e_2) \mid (e_1\,e_3) \\
e_1 \mid e_2 = e_2 \mid e_1 & \emptyset^\star = \Lambda^\star = \Lambda \\
e^\star = \Lambda \mid e^+ & e^+ = e\,e^\star = e^\star\,e \\
e^\star\,e^\star = e^\star & e^{\star\star} = e^\star \\
e = e^\star \Leftrightarrow e = e\,e \wedge e \neq \emptyset & e\,e^\star = e^\star \Leftrightarrow \Lambda \in L(e) \\
(e_1^\star\,e_2^\star)^\star = (e_1 \mid e_2)^\star = (e_1^\star \mid e_2^\star)^\star & \\
(e_1^\star\,e_2)^\star\,(e_1^\star) = (e_1 \mid e_2)^\star = e_1^\star\,(e_2\,(e_1^\star))^\star &
\end{array}$$