

Durée 2h - documents et calculatrices non autorisés

Le barème n'est donné qu'à titre indicatif

1 Logique classique

Nous considérons dans les exercices 1, 2, 4 et 5, les six propositions suivantes :

- (a) Les chevaux sont plus rapides que les chiens.
En anglais, *Horses are faster than dogs.*
- (b) Il existe un lévrier plus rapide que tout lapin.
En anglais, *There exists a greyhound that is faster than all rabbits.*
- (c) Les lévriers sont des chiens.
En anglais, *Greyhounds are dogs.*
- (d) Henri est un cheval.
En anglais, *Henri is a horse.*
- (e) Ronan est un lapin.
En anglais, *Ronan is a rabbit.*
- (f) Si X est plus rapide que Y et Y est plus rapide que Z, alors X est plus rapide que Z.
En anglais, *if X is faster than Y and Y is faster than Z, then X is faster than Z.*

Exercice 1 – Représentation en logique des prédicats du premier ordre – 2.5 points

Traduire en logique des prédicats du premier ordre les six propositions *a*, *b*, *c*, *d*, *e*, *f* en faisant appel aux prédicats unaires *lévrier*(*x*) (lévrier means greyhound in English), *lapin*(*x*) (lapin means rabbit in English), *chien*(*x*) (chien means dog in English) et *cheval*(*x*) (cheval means horse in English), ainsi qu'au prédicat binaire *plus_rapide*(*x*, *y*) (plus_rapide means faster in English).

Exercice 2 – Démonstration avec la règle de résolution – 5 points

1. Mettre les formules *a*, *b*, *c*, *d*, *e*, *f* sous forme de clause, c'est-à-dire de conjonctions, universellement quantifiées, de disjonctions de littéraux.
Pour la formule correspondant à la proposition *b*, il faut pour cela introduire un nouveau symbole de constante, représentant l'individu dont l'existence est indiquée dans l'assertion.
2. Montrer que *Henri est plus rapide que Ronan* découle de *a*, *b*, *c*, *d*, *e*, *f* en utilisant la règle de résolution.

Exercice 3 – Démonstration avec la méthode des tableaux – 3 points

On considère trois variables propositionnelles *p*, *q* et *r*.

Démontrer avec la méthode des tableaux (voir annexe) la validité de la formule suivante :

$$F = (p \wedge q) \vee r \vee (\neg q \wedge \neg r) \vee (\neg p \wedge \neg r)$$

Préciser à chaque étape la règle appliquée et la formule traitée.

2 Prolog

Exercice 4 – Ecriture d'un programme en PROLOG – 3.5 points

1. Formuler les six propositions a, b, c, d, e, f décrite dans la partie 1 sur la logique classique, en Prolog.
Pour la proposition b , il faut pour cela introduire une constante permettant de nommer l'individu dont l'existence est indiquée dans l'assertion.
2. Indiquer comment on peut utiliser Prolog pour prouver que *Henri est plus rapide que Ronan* découle des six propositions a, b, c, d, e, f .

3 Logique de description

Exercice 5 – Représentation en logique de description – 3 points

Représenter les cinq propositions a, b, c, d, e^* décrite dans la partie 1 sur la logique classique, en logique de description \mathcal{ALCT} , en distinguant explicitement la TBox et la ABox.

Cette représentation peut faire appel aux concepts `lévrier`, `lapin`, `chien` et `cheval`, ainsi qu'au rôle `plus_rapide`.

* On ne cherchera pas à représenter la proposition f qui exige des rôles transitifs.

4 Logique modale

Note : Les propositions a, b, c utilisées dans cette partie sont sans lien avec celles du reste de l'examen.

Exercice 6 – Logique modale normale – 1.5 points

Montrez que la formule $\Box\Box\Box\Box(a \vee b \vee c \vee \neg b)$ est valide, dans toute logique modale normale.

Exercice 7 – Modèle de Kripke – 1.5 points

Compléter (si c'est possible, sinon expliquer pourquoi) le modèle de Kripke M comportant trois mondes $\{w_1, w_2, w_3\}$, avec $\pi(a) = \{w_1\}$, $\pi(b) = \{w_2\}$, $\pi(c) = \{w_3\}$ en donnant une relation d'accessibilité R entre mondes qui respecte la sérialité ($\forall w, \exists w' : (w, w') \in R$) et la symétrie ($\forall (w, w') : (w, w') \in R \rightarrow (w', w) \in R$), et telle que les trois formules suivantes soient valides dans ce modèle :

- $a \rightarrow \neg\Diamond b \wedge \neg\Diamond c$
- $b \rightarrow \neg\Diamond b \wedge \Diamond(a \vee c)$
- $c \rightarrow \neg\Diamond c \wedge \Diamond(a \vee b)$

5 Annexe

5.1 Méthode des tableaux sémantiques pour la logique des propositions

La méthode des tableaux sémantiques permet d'établir si un ensemble de fomules logiques est valide, satisfiable ou insatisfiable.

5.1.1 Composantes

La méthode des tableaux est basée sur des règles syntaxiques de décomposition, qui distinguent deux types de formules, nommés α et β .

Nom	Formule α	α_1	α_2
$R_{\neg\neg}$	$\neg\neg\varphi$	φ	φ
R_{\wedge}	$\varphi_1 \wedge \varphi_2$	φ_1	φ_2
$R_{\neg\vee}$	$\neg(\varphi_1 \vee \varphi_2)$	$\neg\varphi_1$	$\neg\varphi_2$
$R_{\neg\rightarrow}$	$\neg(\varphi_1 \rightarrow \varphi_2)$	φ_1	$\neg\varphi_2$
R_{\leftrightarrow}	$\varphi_1 \leftrightarrow \varphi_2$	$\varphi_1 \rightarrow \varphi_2$	$\varphi_2 \rightarrow \varphi_1$

Nom	Formule β	β_1	β_2
R_{\vee}	$\varphi_1 \vee \varphi_2$	φ_1	φ_2
$R_{\neg\wedge}$	$\neg(\varphi_1 \wedge \varphi_2)$	$\neg\varphi_1$	$\neg\varphi_2$
R_{\rightarrow}	$\varphi_1 \rightarrow \varphi_2$	$\neg\varphi_1$	φ_2
$R_{\neg\leftrightarrow}$	$\neg(\varphi_1 \leftrightarrow \varphi_2)$	$\neg(\varphi_1 \rightarrow \varphi_2)$	$\neg(\varphi_2 \rightarrow \varphi_1)$

5.1.2 Satisfiabilité

La recherche d'un modèle pour un ensemble de formules \mathcal{F} par la méthode des tableaux peut être représentée de différentes façons, nous utilisons ici une forme arborescente.

- Initialisation : créer un nœud racine, étiqueté par l'ensemble \mathcal{F} et marqué comme non traité
- Décomposition itérative : choisir un nœud non traité et le marquer comme traité
 - si l'étiquette du nœud contient deux littéraux complémentaires, marquer le nœud comme fermé
 - sinon, si toutes les formules associées au nœud sont des variables propositionnelles, marquer le nœud comme ouvert
 - sinon, choisir une formule F de l'étiquette du nœud
 - si elle est de type α
 - créer un sous-nœud marqué comme non traité
 - lui associer l'étiquette $\mathcal{F} \setminus \{F\} \cup \{\alpha_1, \alpha_2\}$ où α_1 et α_2 sont les formules obtenues par réécriture de F
 - sinon (si elle est de type β)
 - créer deux sous-nœuds marqués comme non traités
 - leur associer respectivement les étiquettes $\mathcal{F} \setminus \{F\} \cup \{\beta_1\}$ et $\mathcal{F} \setminus \{F\} \cup \{\beta_2\}$ où β_1 et β_2 sont les formules obtenues par réécriture de F

Si l'arbre contient une feuille ouverte, alors \mathcal{F} est satisfiable.

Si toutes les feuilles de l'arbre sont fermées, alors \mathcal{F} est insatisfiable.

5.2 Logique de description \mathcal{ALC}

5.2.1 Alphabet

- Concepts atomiques : A, B, C, \dots
- Rôles atomiques : R, S, U, V, \dots
- Symboles : $\{\sqcup, \sqcap, \exists, \forall, \neg, \top, \perp, \cdot\}$
- Instances de concepts : a, b, \dots

5.2.2 Base de connaissances

- TBox - Axiomes terminologiques :
Définitions : $C \equiv D$
Subsumptions : $C \sqsubseteq D$ (se lit C est subsumé par D)
- ABox - Assertions :
Assertions de concepts : $a : C$
Assertions de rôles : $\langle a, b \rangle : R$

5.2.3 Grammaire

concept $:=$ \langle concept atomique \rangle
| \top
| \perp
| \neg \langle concept \rangle
| \langle concept $\rangle \sqcap \langle$ concept \rangle
| \langle concept $\rangle \sqcup \langle$ concept \rangle
| \exists \langle rôle \rangle . \langle concept \rangle
| \forall \langle rôle \rangle . \langle concept \rangle

5.2.4 Sémantique de \exists et \forall

Etant donné une interprétation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, on a :

- $(\exists R.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \exists y, (x, y) \in R^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$
- $(\forall R.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \forall y, (x, y) \in R^{\mathcal{I}} \rightarrow y \in C^{\mathcal{I}}\}$

5.2.5 Extension \mathcal{I}

Grammaire : Si R est un rôle, R^{-1} est un rôle.

Sémantique : $(R^{-1})^{\mathcal{I}} = \{(x, y) \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}, (y, x) \in R^{\mathcal{I}}\}$