Informatique théorique : feuille numéro 1

1 Logique Propositionnelle

1.1 Exercice

P, Q et R désignent trois propositions logiques.

1. Construire les tables de vérité des propositions suivantes :
   (a) $P \Rightarrow (Q \Rightarrow P)$
   (b) $P \Rightarrow (Q \Rightarrow (P \wedge Q))$
   (c) $(P \wedge Q) \Leftrightarrow (\neg(P \Rightarrow \neg Q))$
2. Exprimer sans $\Rightarrow$ ni $\Leftrightarrow$:
   (a) $\neg(P \Leftrightarrow Q)$
   (b) $\neg((P \vee Q) \Rightarrow Q)$
   (c) $\neg(P \Rightarrow (Q \Rightarrow R))$.

1.2 Exercice

Simplifier les propositions suivantes, éventuellement à l'aide d'une table de vérité :

1. $(\neg P \Rightarrow P) \Rightarrow P$
2. $(\neg P \vee Q) \Leftrightarrow (P \Rightarrow Q)$
3. $\neg(P \vee Q) \Leftrightarrow (P \Rightarrow Q)$
4. $(\neg P) \vee (Q \Leftrightarrow (P \Rightarrow Q))$

1.3 Définition : implication sémantique

Soit A une proposition et Γ un ensemble de formules. On dit que Γ implique sémantiquement A si toute valuation qui satisfait simultanément toutes les formules de Γ satisfait également A. Cette propriété est notée $\Gamma \vDash A$. Traditionellement, on écrit Γ sans les accolades utilisées pour les ensembles.

Si Γ est vide, on dit que A est *valide*, et on écrit $\vDash A$. On dit aussi que A est une *tautologie*.

1.4 Exercice

Vérifier si les implications sémantiques ci-dessous sont vraies :

1. $P \vee Q, P \Rightarrow Q, Q \Leftrightarrow R \vDash Q \wedge R$
2. $P \vee Q, \neg Q, P \Rightarrow R \vDash R$
3. $P \Rightarrow \neg P, P \vDash Q$
4. $P \Rightarrow Q \vDash P \wedge Q$

5. $P \Rightarrow Q \vDash P \vee Q$
6. $\neg(P \Rightarrow Q) \vDash P \wedge \neg Q$
7. $\vDash (P \Rightarrow Q) \vee (Q \Rightarrow P)$

1.5 Exercice

On reprend l'exemple vu en cours (signalisation ferroviaire) Vérifier si les implications sémantiques ci-dessous sont vraies. On pose

$$\Gamma = \texttt{RA} \vee \texttt{RB},\ \texttt{A2B} \Rightarrow \texttt{RB},\ \texttt{B2A} \Rightarrow \texttt{RA}$$

1. $\Gamma \vDash \neg(\texttt{A2B} \wedge \texttt{B2A})$
2. $\Gamma \vDash (\neg\texttt{RA} \vee \neg\texttt{RB}) \Rightarrow \neg(\texttt{A2B} \wedge \texttt{B2A})$
3. $\Gamma \vDash \texttt{A2B} \Leftrightarrow \neg\texttt{B2A}$
4. $\Gamma \vDash \texttt{RA} \Rightarrow \texttt{B2A}$

1.6 "Méta-raisonnement"

Montrer les énoncés suivants (Γ, A, B, C sont supposés quelconques, on note $\bot$ la proposition toujours fausse). On emploie aussi la notation $\Gamma, A, \ldots$ pour $\Gamma \cup \{A, \ldots\}$.

1. $\Gamma, A \vDash B$ si et seulement si $\Gamma \vDash A \Rightarrow B$
2. $\Gamma, A \vDash \bot$ si et seulement si $\Gamma \vDash \neg A$
3. $\Gamma, \neg A \vDash \bot$ si et seulement si $\Gamma \vDash A$
4. Si $\Gamma, A \vDash B$ et $\Gamma, \neg A \vDash B$, alors $\Gamma \vDash B$

2 Logique du premier ordre

2.1 Un exercice sur les tableaux

On considère dans cet exercice un tableau t de taille n, contenant des nombres entiers. Beaucoup de propriétés sur ce tableau peuvent s'exprimer en logique du premier ordre à l'aide des notations suivantes :

- $t[i]$: la valeur contenue dans la i-ème case de t
- les symboles relationnels : $i < j$, $i \leq j$, $i = j$.
- la constante n (taille du tableau).

Par exemple " Le tableau t n'est pas vide " s'écrira simplement $n > 0$, et "le tableau t est trié par ordre croissant" s'écrira

$$\forall i : \mathbb{N},\ 0 < i < n - 1 \Rightarrow t[i] \leq t[i + 1]$$

On demande d'écrire de façon similaire les propriétés suivante :

1. Toutes les cases de t contiennent la même valeur

2. Il existe au moins deux valeurs différentes contenues dans t
3. Toutes les cases de t contiennent des valeurs différentes 2 à 2
4. i est l'indice de la valeur la plus grande contenue dans t
5. L'entier x apparait dans une case du tableau t
6. Les k premiers éléments de t sont triés (*tri* par insertion)
7. Les k premiers éléments de t sont triés et sont aussi les k plus petits éléments de t (*tri* par sélection)

Considérons un tableau t' de taille n'. Comment exprimer que le tableau t' est le résultat d'un tri (dans l'ordre croissant) du tableau t ?

2.2 Exercice

On reprend l'exemple ferroviaire. Comment pourrait-on prendre en compte le fait qu'au maximum trois trains peuvent circuler dans le même sens sur le tronçon à voie unique ?

3 Quelques "paradoxes" classiques

Les exemples ci-dessous se retrouvent partout (y compris sur internet). Le mot "paradoxe" est classiquement utilisé. Les premier et troisième exemples illustrent le danger de considérer des hypothèses incohérentes. Le deuxième montre que certains énoncés valides peuvent avoir un contenu contre-intuitif.

3.1 Paradoxe du barbier

On considère une ville dont tous les hommes se rasent eux-mêmes si et seulement si le barbier ne les rase pas.

3.1.1

Exprimer ce "contexte" en logique des prédicats.

3.1.2

Montrer qu'on peut montrer à la fois que le barbier se rase lui-même et qu'il ne se rase pas lui-même.

3.1.3

Que peut-on en déduire ?

3.2 Paradoxe du buveur (*)

> Dans tout pub non vide (c'est à dire qu'il y a au moins une personne présente), il existe une personne qui, si elle boit, alors tout le monde boit.

3.2.1

Exprimer l'énoncé ci-dessus en logique des prédicats.

3.2.2

Montrer que cet énoncé est vrai.

3.3 Paradoxe du détecteur de bouclage (**)

On suppose qu'un utilisateur représente toute fonction d'un langage de programmation (par exemple écrit en `C`, mais pourquoi pas un autre langage) par une chaîne de caractères).

Cet utilisateur autoproclamé génial prétend avoir programmé une fonction :

```
int  loop_detector(char *function, char *argument)
```

qui renvoie 1 si l'appel de `function` sur `argument` provoque un bouclage, et 0 sinon.

Considérons la fonction suivante :

```
int unmask (char *s) {
 if (loop_detector (s, s))
 return 1;
 else
   while(true)
    {}
}
```

En considérant les comportements possibles de l'évaluation de `unmask (unmask)`, démasquer le faux génie.