

TD n°17 - Logique

1 Formules propositionnelles

Exercice 1

Pour chaque formule (on a omis des parenthèses) qui suit, dire si elle est tautologique, antilogique ou juste satisfiable en justifiant.

Remarque : plusieurs outils sont disponibles pour les démonstrations : raisonnement par équivalences, par l'absurde, donner une valuation qui satisfait ou qui falsifie. Ne faites pas appel qu'aux tables de vérité!

1. $x_1 \wedge (x_0 \vee \neg x_0) \wedge \neg x_1$
2. $(x_0 \vee \neg x_1) \wedge (\neg x_0 \vee x_2) \wedge (x_1 \vee \neg x_2)$
3. $(x_0 \vee x_1) \wedge (\neg x_0 \vee x_1) \wedge (x_0 \vee \neg x_1) \wedge (\neg x_0 \vee \neg x_1)$
4. $x_0 \vee (x_0 \rightarrow x_1)$
5. $x_1 \vee (x_0 \rightarrow x_1)$
6. $x_0 \wedge (x_0 \rightarrow x_1)$
7. $x_1 \wedge (x_0 \rightarrow x_1)$
8. $(\neg x_0 \rightarrow x_0) \rightarrow x_0$
9. $(x_0 \wedge (x_0 \rightarrow x_1)) \rightarrow x_1$
10. $(x_0 \wedge x_1) \rightarrow x_0$
11. $(x_0 \vee x_1) \rightarrow x_0$
12. $((x_0 \vee \neg x_0) \rightarrow (x_1 \wedge \neg x_1))$
13. $((x_0 \rightarrow x_1) \wedge (\neg x_0 \rightarrow x_1)) \rightarrow x_0$
14. $((x_0 \rightarrow x_1) \wedge (x_0 \rightarrow \neg x_1)) \rightarrow \neg x_0$
15. $(x_0 \wedge \neg x_0) \rightarrow x_1$
16. $(\neg x_0 \rightarrow x_0) \rightarrow \neg x_0$

Exercice 2

Montrer que toute formule logique A utilisant $\top, \perp, \wedge, \vee, \neg, \rightarrow, \leftrightarrow$ et des variables est sémantiquement équivalente à une formule A' qui ne s'écrit que avec \top, \perp, \neg, \vee et des variables.

Exercice 3

On définit informellement la fonction Neg sur les formules utilisant les variables et les connecteurs \neg, \wedge et \vee par " $Neg(A)$ est obtenue à partir de A en ajoutant un \neg devant chaque symbole de variable, et en remplaçant les \wedge par des \vee et des \vee par des \wedge ".

1. Que vaut $Neg(p \wedge (\neg q \vee r))$?
2. Donner une définition inductive de Neg . **Attention** ici on parle de définition inductive de **fonction**, pas d'ensemble.
3. Montrer par induction structurale que pour toute formule A utilisant uniquement les variables et \neg, \wedge et \vee , on a $Neg(A) \equiv \neg A$.

Exercice 4

Montrer les propriétés suivantes. Lesquelles sont des équivalences sémantiques?

1. $P \rightarrow Q \models ((P \wedge R) \rightarrow (Q \wedge R))$
2. $\{P \vee Q, P \rightarrow R, Q \rightarrow R\} \models R$
3. $\{P \rightarrow (Q \rightarrow R), P \rightarrow Q\} \models P \rightarrow R$
4. $P \models ((\neg P) \rightarrow Q)$
5. $(\neg P) \rightarrow P \models P$

Exercice 5

Caractériser la forme normale conjonctive d'une tautologie.

Bonus : caractériser la forme normale disjonctive d'une tautologie (c'est plus dur).

Exercice 6

Donner une FNC et une FND de $\neg(x_1 \wedge \neg(x_2 \vee x_3))$, où x_1, x_2 et x_3 sont trois variables.

Exercice 7

Montrer que pour toute formules P, P', φ et variables x on a si $P \equiv P'$, alors $P[x \setminus \varphi] \equiv P'[x \setminus \varphi]$.

2 Modélisation logique

Exercice 8

Dans la mythologie grecque, l'accès aux Enfers est gardé par Cerbère, un terrible chien à trois têtes. Celui-ci se trouve à l'entrée de trois couloirs qui, soit permettent de rejoindre le monde des vivants, soit conduisent directement aux Enfers.

Après avoir bu sa coupe de ciguë, Socrate se trouve face à Cerbère. Celui-ci, honoré de rencontrer le grand philosophe, veut lui offrir une chance d'éviter la damnation éternelle. Il lui dit alors : « Je vais t'indiquer un des couloirs qui mène au monde des vivants mais, pour mettre à l'épreuve ta grande sagesse, j'énoncerai trois propositions qui seront soit toutes vraies, soit toutes fausses et tu en déduiras le couloir que tu devras suivre ».

- La première tête dit ensuite : « Le premier couloir ainsi que le troisième mènent au monde des vivants ».
- La deuxième tête dit : « Si le deuxième couloir mène au monde des vivants, alors le troisième n'y mène pas ».
- La troisième tête conclut par : « Le premier couloir mène au monde des vivants, par contre le deuxième n'y mène pas ».

Quel couloir Socrate peut-il emprunter pour rejoindre le monde des vivants? On supposera que Cerbère suit les règles qu'il a énoncé.

Traduire ce problème sous forme de problème logique et le résoudre à l'aide d'une table de vérité.

Exercice 9

Le but de cet exercice est d'exprimer les règles du Sudoku comme des formules logiques, dans le but de se ramener à la résolution de SAT pour résoudre le Sudoku.

1. Quel ensemble de variables va-t-on utiliser (préciser ce qu'elles représentent)?

Si votre réponse à la question précédente ne comprend pas $9 \times 9 \times 9$ variables, recommencez la question précédente.

2. Écrire une formule en FNC qui exprime que chaque case contient une et une seule valeur. (on peut séparer "une" et "une seule" en deux conditions)
3. Écrire une formule en FNC qui exprime que chaque ligne contient une seule fois chaque valeur.
4. Écrire une formule en FNC qui exprime que chaque colonne contient une seule fois chaque valeur.
5. Écrire une formule en FNC qui exprime que chaque sous-carré 3×3 contient une seule fois chaque valeur.
6. On note I l'ensemble des triplets (r, c, k) des cases (r, c) qui sont pré-remplies avec une valeur k . Écrire une formule en FNC qui exprime les contraintes sur les cases pré-remplies.
7. En déduire une formule φ exprimant le problème du sudoku. (On pourra donner des noms à toutes les formules écrites précédemment et ne pas les recopier)
8. Si la formule est satisfiable, et que la méthode employée nous renvoie la valuation correspondante, comment en déduit-on la solution du Sudoku?

3 Manipulation de formules en Ocaml

On utilise le type suivant pour les formules (on remarque qu'il s'agit d'une représentation arborescente) :

```
type formule =
| Top
| Bot
| Var of string
| Non of formule
| Et of formule * formule
| Ou of formule * formule
| Implique of formule * formule
| Equivalent of formule * formule
;;
```

Exercice 10

Écrire une fonction `remplace : formule -> string -> formule -> formule` qui prend en entrée P , x et φ et calcule $P[x \setminus \varphi]$.

Exercice 11

Écrire une fonction `arbre_to_expr : formule -> string` transformant une formule sous forme arborescente en sa forme de mot correctement parenthésée. On pourra faire libre usage de l'opérateur de concaténation des chaînes de caractères en CAML, noté \wedge .

Par exemple, la fonction doit transformer `Ou(Var(x), Et(Top, Bot))` en la chaîne "x ou (vrai et faux)".

4 Formules du premier ordre

Exercice 12

On considère la signature (X, S_f, S_p) :

$$X = \{x, y, z\} \quad S_f = \{(c, 0), (f, 1), (g, 2)\} \quad S_p = \{(P, 1), (Q, 2), (R, 3)\}$$

et les expressions suivantes :

1. $g(f(c), g(x, f(f(y))))$,
2. $f(f(c), f(c, f(c)))$,
3. $Q(f(x), P(y))$,
4. $R(f(c), f(f(c)), y)$,
5. $P(f(x)) \wedge \exists y(Q(y, x) \rightarrow R(y, x, f(x)))$,
6. $P(f(x)) \vee R(g(x, y), f(z), y)$,
7. $\forall x \exists y g(x, f(y))$,
8. $\exists y \forall x Q(g(x), f(y, g(c)))$,
9. $\forall x \exists y \forall z (Q(x, f(y), g(c, z)) \vee P(x))$,

Pour chaque expression dites s'il s'agit d'un terme, d'une formule atomique, d'une formule du premier ordre, aucun des précédents.

Si la formule peut être classée, dessinez l'arbre de syntaxe correspondant. Sinon expliquez pourquoi elle ne peut pas être classée.

Exercice 13

On considère l'ensemble des variables $X = \{x, y, z\}$ et les formules suivantes :

$$\varphi_1 = (\forall x \exists z. f(x, z)) \rightarrow (\exists x \forall y. r(x, y, z))$$

$$\varphi_2 = (\forall x. p(x) \wedge \forall x. f(x)) \rightarrow \forall x. (p(x) \wedge f(x))$$

$$\varphi_3 = \forall x. ((\exists x. g(f(x), a) \vee h(x, x)) \wedge ((\forall y \exists x. q(x, y)) \vee (\exists z. p(z, y))))$$

1. Pour chaque formule, donnez la signature (ensemble des fonctions et prédicats) sur laquelle est écrite.
2. Listez les termes et les formules atomiques qui apparaissent dans chaque formule.