

## TD n°17 - Corrigé

### 1 Formules propositionnelles

#### Exercice 1

On notera  $\varphi_i$  la formule de la question  $i$ .

1. Pour toute valuation  $v$ , on a  $[[\varphi_1]]_v = \text{et}([x_1 \wedge \neg x_1]]_v, [x_0 \vee \neg x_0]]_v)$ . Or  $x_1 \wedge \neg x_1$  est une antilogie et la conjonction de faux et d'autre chose est toujours faux, donc  $\varphi_1$  est une antilogie.
2. On dessine la table de vérité (en fait seule les deux premières lignes sont nécessaires. On conclut que  $\varphi_2$  est satisfiable mais pas une tautologie.

$[[x_0]]_v$	$[[x_1]]_v$	$[[x_2]]_v$	$[[\varphi_2]]_v$
V	V	V	V
V	V	F	F
V	F	V	F
V	F	F	F
F	V	V	F
F	V	F	F
F	F	V	F
F	F	F	V

3. On teste toutes les valuations possibles  $v_1$  qui associe toujours V,  $v_2$  qui associe V à  $x_0$  et F à  $x_1$ ,  $v_3$  qui associe F à  $v_1$  et V à  $v_2$  et  $v_4$  qui associe F aux deux variables.  
On a alors  $[[\varphi_3]]_{v_1} = [[\varphi_3]]_{v_2} = [[\varphi_3]]_{v_3} = [[\varphi_3]]_{v_4} = F$ .  $\varphi_3$  est une antilogie.
4. On remarque que si  $x_0$  est satisfait, alors la formule aussi. Si  $x_0$  n'est pas satisfiable, alors  $x_0 \rightarrow x_1$  est satisfait et la formule aussi. Ainsi  $\varphi_3$  est une tautologie.
5. Soit  $v_1$  associant V à  $x_1$  et V à  $x_0$ , alors  $[[\varphi_5]]_{v_1} = V$ .  
Soit  $v_2$  associant F à  $x_1$  et V à  $x_0$ , alors  $[[\varphi_4]]_{v_2} = F$ .  
On en conclut que  $\varphi_6$  est satisfiable mais pas une tautologie.
6. Soit  $v_1$  associant V à  $x_1$  et V à  $x_0$ , alors  $[[\varphi_6]]_{v_1} = V$ .  
Soit  $v_2$  associant V à  $x_1$  et F à  $x_0$ , alors  $[[\varphi_6]]_{v_2} = F$ .  
On en conclut que  $\varphi_6$  est satisfiable mais pas une tautologie.
7. Soit  $v_1$  associant V à  $x_1$  et F à  $x_0$ , alors  $[[\varphi_7]]_{v_1} = V$ .  
Soit  $v_2$  associant F à  $x_1$  et V à  $x_0$ , alors  $[[\varphi_7]]_{v_2} = F$ .  
On en conclut que  $\varphi_7$  est satisfiable mais pas une tautologie.
8. On raisonne par équivalences :

$$\begin{aligned} x_0 \vee (x_0 \rightarrow x_0) &\equiv (x_0 \vee x_0) \rightarrow x_0 \\ &\equiv x_0 \rightarrow x_0 \\ &\equiv \neg x_0 \vee x_0 \end{aligned}$$

Cette dernière expression est une tautologie, donc  $\varphi_8$  aussi.

9. On va simplifier et conclure.

$$\begin{aligned} (x_0 \wedge (x_0 \rightarrow x_1)) \rightarrow x_1 &\equiv (x_0 \wedge (\neg x_0 \vee x_1)) \rightarrow x_1 \text{ en utilisant la définition de } \rightarrow \\ &\equiv ((x_0 \wedge \neg x_0) \vee (x_0 \wedge x_1)) \rightarrow x_1 \text{ en utilisant De Morgan} \\ &\equiv (\perp \vee (x_0 \wedge x_1)) \rightarrow x_1 \text{ en remarquant une antilogie} \\ &\equiv (x_0 \wedge x_1) \rightarrow x_1 \\ &\equiv (\neg x_0 \wedge \neg x_1) \vee x_1 \text{ en transformant l'implication et en développant} \\ &\equiv (\neg x_0 \vee x_1) \wedge (\neg x_1 \vee x_1) \text{ en développant} \\ &\equiv (\neg x_0 \vee x_1) \wedge \top \text{ en remarquant une tautologie} \\ &\equiv \neg x_0 \vee x_1 \end{aligned}$$

La formule finale est satisfiable ( $v(x_0) = F$  et  $v(x_1) = V$ ) et non tautologique ( $v(x_0) = V$  et  $v(x_1) = F$ ).

10. Si  $(x_0 \wedge x_1)$  est évalué à V pour une valuation  $v$ , alors  $v(x_0) = V$  donc la formule est évaluée à V.

Si  $(x_0 \wedge x_1)$  est évalué à F, alors la formule est évaluée à V.

$\varphi_{10}$  est une tautologie.

11. On considère  $\begin{cases} v_1(x_0) = V \\ v_1(x_1) = F \end{cases}$  qui satisfait  $\varphi_{11}$  et  $\begin{cases} v_1(x_0) = F \\ v_1(x_1) = V \end{cases}$  qui ne satisfait pas  $\varphi_{11}$ .  $\varphi_{11}$  est satisfiable mais pas une tautologie.

12. À gauche de la flèche on remarque une tautologie et à droite une antilogie.

Or, quelque soit  $v$ ,  $[[\varphi_{12}]]_v = \text{implique}([x_0 \vee \neg x_0]_v, [x_1 \wedge \neg x_1]_v) = \text{implique}(V, F) = F$

Conclusion :  $\varphi_{12}$  est une antilogie.

$[[x_0]]_v$	$[[x_1]]_v$	$[[x_0 \rightarrow x_1]]_v$	$[[\neg x_0 \rightarrow x_1]]_v$	$[[x_0 \rightarrow x_1] \wedge (\neg x_0 \rightarrow x_1)]_v$	$[[\varphi_{13}]]$
V	V	V	V	V	V
V	F	F	V	F	V
F	V	V	V	V	F
F	F	V	F	F	V

On fait manifestement face à une formule satisfiable non tautologique.

$[[x_0]]_v$	$[[x_1]]_v$	$[[x_0 \rightarrow x_1]]_v$	$[[x_0 \rightarrow \neg x_1]]_v$	$[[x_0 \rightarrow x_1] \wedge (x_0 \rightarrow \neg x_1)]_v$	$[[\varphi_{14}]]$
V	V	V	F	F	V
V	F	F	V	F	V
F	V	V	V	V	V
F	F	V	V	V	V

On fait manifestement face à une tautologie.

15. On a à gauche une antilogie, donc l'implication est toujours vraie, peut importe l'évaluation de  $x_1$  et  $x_0$ .

16. On raisonne par équivalences :

$$\begin{aligned} (\neg x_0 \rightarrow x_0) \rightarrow \neg x_0 &\equiv (\neg \neg x_0 \vee x_0) \rightarrow \neg x_0 \\ &\equiv x_0 \rightarrow \neg x_0 \\ &\equiv \neg x_0 \vee \neg x_0 \\ &\equiv \neg x_0 \end{aligned}$$

La formule  $\varphi_{16}$  est donc, comme  $\neg x_0$  satisfiable mais non tautologique.

### 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.

On raisonne par induction structurelle sur la structure de la formule.

- Si  $A = \top$ ,  $A = \perp$  ou  $A = x$ , on prend  $A' = A$ , donc on a  $A \equiv A'$ .
- Si  $A = \neg B$  avec  $B$  qui vérifie l'hypothèse d'induction, c'est à dire qu'on peut obtenir  $B \equiv B'$  avec  $B'$  comme voulu, alors  $A' = \neg B'$  vérifie  $A \equiv A'$  et est de la bonne forme.
- Si  $A = B \vee C$ , en notant  $B'$  et  $C'$  les formules obtenues par application de l'hypothèse d'induction sur  $B$  et  $C$ , on a  $A \equiv B' \vee C'$ .
- Si  $A = B \wedge C$ , en notant  $B'$  et  $C'$  les formules obtenues par application de l'hypothèse d'induction sur  $B$  et  $C$ , on a  $A \equiv \neg(\neg B' \vee \neg C')$  en utilisant les lois de De Morgan.
- Si  $A = B \rightarrow C$  ou  $A = B \leftrightarrow C$ , alors on utilise les équivalences classiques pour se ramener à  $\neg B \vee C$  ou  $(\neg B \vee C) \wedge (B \vee \neg C)$  et on fait disparaître les  $\wedge$  comme dans le cas précédent.

### 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. En appliquant la définition,  $\neg p \vee (\neg \neg q \wedge r)$ . On notera que c'est équivalent à  $\neg p \vee (q \wedge r)$

2. On définit  $Neg$  sur toutes les formes de formules possibles :

$$\begin{aligned}
 Neg(x) &= \neg x \\
 Neg(\neg P) &= \neg Neg(P) \\
 Neg(P \wedge Q) &= Neg(P) \vee Neg(Q) \\
 Neg(P \vee Q) &= Neg(P) \wedge Neg(Q)
 \end{aligned}$$

3. On rappelle que  $A = A' \implies A \equiv A'$ .

Comme demandé, on procède par induction structurale sur la structure de la formule

- Si  $A = x$  avec  $x$  une variable, alors  $Neg(A) = \neg x = \neg A$ .
- Si  $A = P \wedge Q$ , alors  $Neg(A) = Neg(P) \vee Neg(Q)$ . On applique l'hypothèse d'induction :  $Neg(P) \equiv \neg P$  et  $Neg(Q) \equiv \neg Q$ .  
Alors on a  $Neg(A) \equiv \neg P \vee \neg Q \equiv \neg(P \wedge Q) = \neg A$ , en utilisant une loi de De Morgan.
- Si  $A = P \vee Q$ , alors  $Neg(A) = Neg(P) \wedge Neg(Q)$ . On applique l'hypothèse d'induction :  $Neg(P) \equiv \neg P$  et  $Neg(Q) \equiv \neg Q$ .  
Alors on a  $Neg(A) \equiv \neg P \wedge \neg Q \equiv \neg(P \vee Q) = \neg A$ , en utilisant l'autre loi de De Morgan.
- Si  $A = \neg P$  alors  $Neg(A) = \neg Neg(P)$ , or par hypothèse d'induction  $Neg(P) \equiv \neg P$ , d'où  $Neg(A) \equiv \neg \neg P = \neg A$ .

**Exercice 4**

Montrer les propriétés suivantes. Lesquelles sont des équivalences ?

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.  $(\neg P) \rightarrow P \models P$

1. Pour cette première question on dessine une table de vérité

$[[P]]_v$	$[[Q]]_v$	$[[R]]_v$	$[[P \rightarrow Q]]_v$	$[[((P \wedge R) \rightarrow (Q \wedge R))]]_v$
V	V	V	V	V
V	V	F	V	V
V	F	V	F	F
V	F	F	F	V
F	V	V	V	V
F	V	F	V	V
F	F	V	V	V
F	F	F	V	V

On voit que tout modèle de la formule de gauche est un modèle de la formule de droite. En revanche l'inverse est faux, comme on peut le voir sur la quatrième ligne.

2. On note  $\Gamma = \{P \vee Q, P \rightarrow R, Q \rightarrow R\}$

Soit  $v$  un modèle de  $\Gamma$ . On a plusieurs possibilités :

Soit  $v$  n'est pas un modèle de  $P$ , auquel cas c'est un modèle de  $Q$  puisque c'est un modèle de  $P \vee Q$ . On en déduit que  $v$  est un modèle de  $R$  car c'est un modèle de  $Q \rightarrow R$  et de  $Q$ .

Soit  $v$  est modèle de  $P$ , donc modèle de  $R$  puisqu'on ne peut pas évaluer  $P \rightarrow R$  en évaluant  $P$  à vrai et  $R$  à faux.

Ce n'est pas une équivalence en général, supposons l'existence d'une valuation  $v$  telle que  $[[P]]_v = F$ ,  $[[Q]]_v = F$  et  $[[R]]_v = V$ , alors c'est un modèle de  $R$  mais pas de  $\Gamma$ .

Bien sûr il est possible qu'un tel  $v$  n'existe pas. Il existe donc des  $P, Q$  et  $R$  tels que c'est une équivalence.

3. Soit  $v$  un modèle du terme gauche. Supposons par l'absurde qu'il n'est ni modèle de  $\neg P$ , ni modèle de  $R$ . Alors pour qu'il soit modèle de  $(P \rightarrow Q)$  il doit être modèle de  $Q$  et pour être modèle de  $P \rightarrow (Q \rightarrow R)$  il doit être modèle de  $\neg Q$ . C'est absurde.

Considérons  $v$  tel que  $[[R]]_v = V$ ,  $[[P]]_v = V$  et  $[[Q]]_v = F$ , alors c'est un modèle du terme de droite mais pas du terme de gauche.

4.  $(\neg P) \rightarrow Q \equiv P \vee Q$ . Donc si  $v$  est un modèle de  $P$ , alors c'est un modèle du terme droit.

L'inverse est faux, il suffit de considérer un modèle  $v$  du terme droit qui vérifie  $[[P]]_v = F$  et  $[[Q]]_v = V$ .

5. On remarque très facilement  $(\neg P) \rightarrow P \equiv P$ .

### Exercice 5

#### Forme normale conjonctive

Pour la forme normale conjonctive, si  $C_1 \wedge C_2 \wedge \dots \wedge C_n$  est une tautologie si et seulement si pour tout  $i \in \llbracket 1, n \rrbracket$ ,  $C_i$  est une tautologie.

Montrons qu'une clause est une tautologie si et seulement si elle contient comme littéraux une variable et son contraire.

Le sens réciproque est évident.

Pour le sens direct, on raisonne par l'absurde, soit  $C = l_1 \vee l_2 \vee \dots \vee l_m$  une clause qui est une tautologie. On suppose qu'on a pas.  $l_i \equiv \neg l_j$

On considère alors une valuation  $v$  qui à chaque variable associe vrai si le littéral présent dans  $C$  est négatif et faux sinon (s'il existe d'autres variables qui n'apparaissent pas dans  $C$ , on les fixe toutes à vrai.) Ce processus est valide puisqu'on a dit qu'une même variable n'apparaissait que positivement ou que négativement.

Par construction  $v$  n'est modèle d'aucun des littéraux, donc il ne peut pas être modèle de  $C$ , ce qui est absurde puisqu'il s'agit d'une tautologie.

**Conclusion** : la forme normale conjonctive d'une tautologie est formée de clauses qui contiennent toutes au moins un littéral et son contraire.

#### Forme normale disjonctive

Pour la forme normale disjonctive c'est plus compliqué : on montre qu'une formule  $\phi$  en FND est une tautologie si et seulement si en sélectionnant un littéral dans chaque clause disjonctive, on obtient toujours aux moins deux littéraux opposés.

Supposons que  $D_1 \vee D_2 \vee \dots \vee D_n$  est une tautologie et que peu importe comment on les sélectionne, un ensemble  $\{l_1, l_2, \dots, l_n\}$ , avec  $\forall i, l_i \in D_i$ , ne contient jamais deux littéraux qui sont opposés (ie  $\forall i, j, l_i \neq \neg l_j$ ). Alors on définit une valuation  $v$  telle que pour chaque variable  $x$  apparaissant dans un  $l_j$  on a  $v(x) = V$  si  $l_j = \neg x$  et  $v(x) = F$  si  $l_j = x$ .

On remarque alors que peu importe la clause disjonctive  $D_j$  considérée,  $v$  n'en est pas un modèle puisque qu'il ne rend pas  $l_j$  vrai. Ainsi  $\Phi$  n'est pas une tautologie, ce qui est absurde.

Réciproquement on suppose que  $\Phi$  n'est pas une tautologie, donc il existe une valuation  $v$  qui contredit  $\Phi$ , donc qui contredit chaque  $D_j$ . Ainsi dans chaque  $D_j$  on peut sélectionner au moins un  $l_j$  qui n'est pas satisfait par  $v$ . L'ensemble  $\{l_1, \dots, l_n\}$  ne peut pas contenir deux littéraux opposés puisque, peu importe la variable  $x$ ,  $v$  ne peut pas rendre faux à la fois  $x$  et  $\neg x$ .

On a conclut. On remarquera que cela implique que la FND d'une tautologie contient toujours une clause disjonctive avec uniquement des littéraux négatifs et une autre clause disjonctive avec uniquement des littéraux positifs.

### Exercice 6

En utilisant une fois les lois de De Morgan, on a  $\neg(x_1 \wedge \neg(x_2 \vee x_3)) \equiv \neg x_1 \vee \neg\neg(x_2 \vee x_3) \equiv \neg x_1 \vee x_2 \vee x_3$ .

Cette dernière forme est à la fois une FND et une FNC.

### Exercice 7

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

On note  $\mathcal{V}$  l'ensemble des variables existantes (celui qui est dans la définition inductive des propositions).

Cet exercice nécessite un peu de subtilité dans la manipulation des variables. On notera  $Var(\psi)$  les variables qui apparaissent dans une formule  $\psi$  quelconque.

Soit  $\psi$  un formule,  $y$  une variable n'apparaissant pas dans  $\psi$ ,  $v : \mathcal{V} \mapsto \{V, F\}$  une valuation et  $v' = v|_{\mathcal{V} \setminus \{y\}}$ . Alors  $[[\psi]]_v = [[\psi]]_{v'}$ . Ainsi l'évaluation dans le contexte  $v$  de  $\psi$  revient à son évaluation dans le contexte  $v|_{Var(\psi)}$ .

De plus, comme on remplace  $x$  par  $\varphi$ , si  $\varphi$  contient la variable  $x$ , alors on peut la renommer  $x'$ . En effet la variable  $x$  qui disparaît de  $P$  et  $P'$  et celle qui est ajoutée par  $\varphi$  n'existeront pas en même temps dans les formules  $P, P', P[x \setminus \varphi], P'[x \setminus \varphi]$ , donc ne sont pas corrélées. On peut donc supposer que  $\varphi$  ne contient pas  $x$ .

Enfin on a dit que  $P \equiv P'$  revient à "pour toute valuation  $v$ ,  $[[P]]_v = [[P']]_v$ ". Ceci n'a de sens que si l'espace de départ de  $v$  est au moins  $Var(P) \cup Var(P')$ .  $P$  et  $P'$  ne contiennent pas forcément les mêmes variables. L'égalité  $[[P[x \setminus \varphi]]]_v = [[P'[x \setminus \varphi]]]_v$ , elle, n'a de sens que si l'espace de départ de  $v$  est au moins  $(Var(P) \cup Var(P') \cup Var(\varphi)) \setminus \{x\}$ . ( $x$  n'existant pas dans les formules comparées)

Montrons désormais un **lemme** : pour  $Q, \psi$  des formules,  $x$  une variable apparaissant dans  $Q$  et  $v : (Var(Q) \cup Var(\psi)) \setminus \{x\} \mapsto \{V, F\}$  une valuation, si on définit  $v' : Var(Q) \mapsto \{V, F\}$  par  $v'(y) = v(y)$  si  $y \neq x$  et  $v'(x) = \llbracket \psi \rrbracket_v$ , alors  $\llbracket Q[x \setminus \psi] \rrbracket_v = \llbracket Q \rrbracket_{v'}$ .

Montrons le par induction structurelle sur la structure de  $Q$ .

- Si  $Q$  est restreint à une variable, cette variable est  $x$ . Donc  $Q = x$  et  $Q[x \setminus \psi] = \psi$ .  
Dans ce cas  $\llbracket Q \rrbracket_{v'} = \llbracket \psi \rrbracket_v = \llbracket Q[x \setminus \psi] \rrbracket_v$ .
- $Q$  ne peut pas être  $\top$  ou  $\perp$ .
- Si  $Q = \neg P$  et qu'on suppose l'hypothèse d'induction sur  $P$ . Alors  $Q[x \setminus \psi] = \neg P[x \setminus \psi]$ , donc  $\llbracket Q[x \setminus \psi] \rrbracket_v = \text{non}(\llbracket P[x \setminus \psi] \rrbracket_v) = \text{non}(\llbracket P \rrbracket_{v'}) = \llbracket Q \rrbracket_{v'}$ .
- Si  $Q = Q_1 \wedge Q_2$  et qu'on suppose l'hypothèse d'induction sur  $Q_1$  et  $Q_2$ . Alors  $Q[x \setminus \psi] = Q_1[x \setminus \psi] \wedge Q_2[x \setminus \psi]$ , donc  $\llbracket Q[x \setminus \psi] \rrbracket_v = \text{and}(\llbracket Q_1[x \setminus \psi] \rrbracket_v, \llbracket Q_2[x \setminus \psi] \rrbracket_v) = \text{et}(\llbracket Q_1 \rrbracket_{v'}, \llbracket Q_2 \rrbracket_{v'}) = \llbracket Q \rrbracket_{v'}$ .
- Les autres cas se font de la même manière.

Passons à l'exercice. Soit  $v : (Var(P) \cup Var(P') \cup Var(\varphi)) \setminus \{x\} \mapsto \{V, F\}$ . On veut montrer que  $\llbracket P[x \setminus \varphi] \rrbracket_v = \llbracket P'[x \setminus \varphi] \rrbracket_v$ .

On définit  $v' : Var(P) \cup Var(P') \mapsto \{V, F\}$  par  $v'(y) = v(y)$  si  $y \neq x$  et  $v'(x) = \llbracket \varphi \rrbracket_v$ .

Alors d'après le lemme  $\llbracket P \rrbracket_{v'} = \llbracket P[x \setminus \varphi] \rrbracket_v$  et  $\llbracket P' \rrbracket_{v'} = \llbracket P'[x \setminus \varphi] \rrbracket_v$ . Or  $P = P'$ , donc  $\llbracket P \rrbracket_{v'} = \llbracket P' \rrbracket_{v'}$ . Finalement  $\llbracket P[x \setminus \varphi] \rrbracket_v = \llbracket P'[x \setminus \varphi] \rrbracket_v$ . Ceci vaut pour tout  $v$ , donc  $P[x \setminus \varphi] \equiv P'[x \setminus \varphi]$ .

## 2 Modélisation logique

### Exercice 8

On traduit les propositions, on fait la table de vérité et on remarque qu'il y a une valuation qui rend vraies toutes les propositions (Cerbère ne ment pas) et une valuation qui rend fausses toutes les propositions (Cerbère a menti). Ces deux valuations on en commun d'associer V à la variable représentant le fait que la 3e porte mène vers le monde des vivants.

Donc Socrate doit prendre la troisième porte pour survivre.

### Exercice 9

1. Ici on considèrera les variables  $x_{r,c,v}$  qui valent vrai si à la case  $(r, c)$  il y a la valeur  $v$ .
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)
 
$$\bigwedge_{r \in \llbracket 0,8 \rrbracket} \bigwedge_{c \in \llbracket 0,8 \rrbracket} ((x_{r,c,1} \wedge \neg x_{r,c,2} \wedge \dots \wedge \neg x_{r,c,9}) \vee (\neg x_{r,c,1} \wedge x_{r,c,2} \wedge \dots \wedge \neg x_{r,c,9}) \vee \dots \vee (\neg x_{r,c,1} \wedge \neg x_{r,c,2} \wedge \dots \wedge x_{r,c,9}))$$
3. Écrire une formule en FNC qui exprime que chaque ligne contient une seule fois chaque valeur.
 
$$\bigwedge_{r \in \llbracket 0,8 \rrbracket} \bigwedge_{v \in \llbracket 1,9 \rrbracket} \bigvee_{c \in \llbracket 0,8 \rrbracket} (x_{r,c,v})$$
 Remarque, ici on a pas besoin d'un XOR, car on a déjà dit qu'une case n'avait qu'une et une seule valeur, donc si chaque couleur va dans au moins une colonne, comme il y a autant de couleurs que de colonnes, chaque couleur va dans exactement une colonne.
4. Écrire une formule en FNC qui exprime que chaque colonne contient une seule fois chaque valeur.
 
$$\bigwedge_{c \in \llbracket 0,8 \rrbracket} \bigwedge_{v \in \llbracket 1,9 \rrbracket} \bigvee_{r \in \llbracket 0,8 \rrbracket} (x_{r,c,v})$$
5. Écrire une formule en FNC qui exprime que chaque sous-carré  $3 \times 3$  contient une seule fois chaque valeur.
 
$$\bigwedge_{a \in \{0,1,2\}} \bigwedge_{b \in \{0,1,2\}} \bigwedge_{v \in \llbracket 1,9 \rrbracket} \bigvee_{r \in \{a*3+0, a*3+1, a*3+2\}} \bigvee_{c \in \{b*3+0, b*3+1, b*3+2\}} (x_{r,c,v})$$
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.
 
$$\bigwedge_{r,c,k \in I} x_{r,c,v}$$
7. En déduire une formule  $\varphi$  exprimant le problème du sudoku. (On pourra donner des noms à toutes les formules écrites précédemment et ne pas les recopier)  
En notant  $\varphi_i$  la formule écrite à la question  $i$ , la formule totale est  $\varphi_2 \wedge \varphi_3 \wedge \varphi_4 \wedge \varphi_5 \wedge \varphi_6$ .
8. Si la formule est satisfiable, et que la méthode employée nous renvoie la valuation correspondante, comment en déduit-on la solution du Sudoku?  
Si  $x_{r,c,v}$  est vrai, cela signifie que la case  $r, c$  contient  $v$  dans la solution trouvée.

### 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  $\varphi$  et calcule  $P[x \setminus \varphi]$ .

```
let rec replace f x phi = match f with
| Var(y) when y=x -> phi
| Non(f') -> Non(replace f' x phi)
| Et(f1,f2) -> Et(replace f1 x phi, replace f2 x phi)
| Ou(f1,f2) -> Ou(replace f1 x phi, replace f2 x phi)
| Implique(f1,f2) -> Implique(replace f1 x phi, replace f2 x phi)
| Equivalent(f1,f2) -> Equivalent(replace f1 x phi, replace f2 x phi)
|_ -> f
;;
```

#### 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 `^`.

```
let rec arbre_to_expr f = match f with
| Top -> "vrai"
| Bot -> "faux"
| Var(y) -> y
| Non(f') -> "non (" ^ (arbre_to_expr f') ^ ")"
| Et(f1,f2) -> "(" ^ (arbre_to_expr f1) ^ " et " ^ (arbre_to_expr f2) ^ ")"
| Ou(f1,f2) -> "(" ^ (arbre_to_expr f1) ^ " ou " ^ (arbre_to_expr f2) ^ ")"
| Implique(f1,f2) -> "(" ^ (arbre_to_expr f1) ^ " implique " ^ (arbre_to_expr f2) ^ ")"
| Equivalent(f1,f2) -> "(" ^ (arbre_to_expr f1) ^ " est équivalent a " ^ (arbre_to_expr f2) ^ ")"
;;
```

### 4 Formules du premier ordre

#### Exercice 12

- terme,
- mal formé,  $f$  est d'arité 1 et pas 2,
- mal formé,  $Q$  prend en argument  $P$  de quelque chose,
- formule atomique (et formule),
- formule,
- formule,
- mal formé, c'est le résultat de  $g$  qui est quantifié et  $g$  n'est pas un prédicat,
- mal formé à cause des arités de  $f$  et  $g$ ,
- mal formé à cause de l'arité de  $Q$ ,

#### Exercice 13

Pour  $\varphi_1$ ,  $S_f = \{ \}$  et  $S_p = \{ (f, 2), (r, 3) \}$ . Les seuls termes apparaissant sont des variables et les formules atomiques sont  $r(x, y, z)$  et  $f(x, z)$ .

Pour  $\varphi_2$ ,  $S_f = \{ \}$  et  $S_p = \{ (f, 1), (p, 1) \}$ . Les seuls termes apparaissant sont des variables et les formules atomiques sont  $p(x)$  et  $f(x)$ .

Pour  $\varphi_3$ ,  $S_f = \{ (a, 0), (f, 1) \}$  et  $S_p = \{ (g, 2), (h, 2), (q, 2), (p, 2) \}$ . Les termes sont les variables,  $a$  et  $f(x)$ . Les formules atomiques sont  $g(f(x), a)$ ,  $h(x, x)$ ,  $q(x, y)$  et  $p(z, y)$ .