1 Généralités

1) Déterminer les tables de vérités des formules suivantes :

   a) $v_1 \wedge (\neg v_2 \vee v_3)$
   b) $(v_1 \vee (v_2 \wedge v_3)) \wedge (\neg v_1 \vee (\neg v_2 \wedge \neg v_3))$
   c) $v_1 \Rightarrow (v_2 \Rightarrow v_1)$
   d) $(v_1 \vee (v_2 \Rightarrow v_3)) \Rightarrow (v_1 \vee v_2 \vee v_3)$.

2) Dire si les formules suivantes sont des tautologies, des antilogies et si elles sont satisfiables.

   a) $(v_1 \vee v_2) \vee \neg(v_2 \vee v_1)$
   b) $(v_1 \text{ xor } v_2) \text{ xor } \neg v_1$.
   c) $v_1 \Rightarrow (v_2 \text{ xor}(v_3 \vee \neg v_2))$.
   d) $(p_1 \Rightarrow p_2) \Rightarrow \big((p_2 \Rightarrow p_3) \Rightarrow (p_1 \Rightarrow p_3)\big)$

3) Démontrer que, si p_1 et $p_1 \Rightarrow p_2$ sont des tautologies, alors p_2 est une tautologie.

4) Soit p une formule. On appelle longueur de p le nombre de caractères nécessaires à écrire la formule en comptant que :

   - Les variables propositionnelles et les constantes V et F comptent pour un caractère.
   - On compte les parenthèses ; on rappelle que l'on utilise des parenthèses autour de $(p_1 \wedge p_2)$ et de $(p_1 \vee p_2)$ mais pas pour $\neg p$.

   Par exemple la formule $(((v_1 \wedge F) \vee \neg(v_3 \vee v_4)) \wedge \neg v_5)$ compte pour 19 caractères.

   a) On utilise l'implémentation suivante des formule logique.

   ```
   type formule =
     | V
     | F
     | Var of int
     | Non of formule
     | Ou of formule * formule
     | Et of formule * formule;;
   ```

   Écrire une fonction longueur.

   b) On désigne par M_n (resp m_n) la longueur maximale (resp. minimale) d'une formule de hauteur n. Déterminer des formules de récurrences définissant (M_n) et (m_n) et en déduire le terme général de ces suites.

5) On appelle connecteur de Sheffer et on note | le connecteur défini par

$$p_1 | p_2 \equiv (\neg p_1 \vee \neg p_2)$$

   où p_1 et p_2 sont des formules logiques. *Ce connecteur s'appelle aussi NAND.*

a) Soit p une formule logique, montrer que $\neg p$ et $p|p$ sont équivalentes.

b) Soit p_1 et p_2 des formules logiques, déterminer de même une formule équivalente à $(p_1 \vee p_2)$ ne faisant intervenir que p_1, p_2 et $|$.

c) En faire de même pour $(p_1 \wedge p_2)$.

d) En déduire que toute formule logique peut s'écrire uniquement à l'aide des constantes V et F, des variables propositionnelles et du connecteur de Scheffer. *On dit que* $|$ *est un système complet de connecteurs.*

6) Soit `formule` le type permettant de coder des formules logiques de l'exercice 4.

On suppose qu'on dispose d'une fonction `sat : formule -> bool` permettant de savoir si une formule est satisfiable. Montrer qu'il est possible d'utiliser cette fonction pour résoudre les problèmes suivants. On écrira les fonctions correspondantes.

a) Déterminer si une formule est une antilogie.

b) Déterminer si une formule est une tautologie.

c) Déterminer si deux formules sont sémantiquement égales.

7) On considère un ensemble non vide de variables propositionnelles.

a) Soit le connecteur ternaire $G(p_1, p_2, p_3)$ défini par :

$$G(p_1, p_2, p_3) = (\neg p_1 \wedge p_2 \wedge p_3) \vee (p_1 \wedge \neg p_2) \vee (\neg p_2 \wedge \neg p_3).$$

Montrer que toute formule logique est équivalente à une expression utilisant uniquement G comme connecteur et une variable propositionnelle (mais pas V et F).

b) Soit $-$ le connecteur binaire défini par :

$$p_2 - p_1 = (\neg p_1 \wedge p_2).$$

Montrer que toute expression booléenne est équivalente à une expression utilisant uniquement le connecteur $-$, les variables propositionnelles et la constante V.

c) Toute expression booléenne est-elle équivalente à une expression utilisant uniquement le connecteur $-$ et les variables propositionnelles

8) Fonctions booléennes

Soit une fonction booléenne $f(x_1, x_2, \ldots, x_i, \ldots, x_n)$ des n variables $x_1, x_2, \ldots, x_n$.
On appelle résidu de f par rapport à x_i (noté f_{x_i}) la fonction des $n-1$ variables x_1, x_2,...,x_{i-1}, x_{i+1},...,x_n qui correspond à une expression logique de f dans laquelle on a remplacé x_i par 1 :

$$f_{x_i} : (x_1, x_2, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) \longmapsto f(x_1, x_2, \ldots, x_{i-1}, 1, x_{i+1}, \ldots, x_n)$$

De même, on appelle résidu de f par rapport à $\bar{x}_i$ (noté $f_{\bar{x}_i}$) la fonction :

$$f_{\overline{x_i}} : (x_1, x_2, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) \longmapsto f(x_1, x_2, \ldots, x_{i-1}, 0, x_{i+1}, \ldots, x_n)$$

a) Démontrer que $f \equiv (x_i \wedge f_{x_i}) \vee (\overline{x_i} \wedge f_{\overline{x_i}})$

b) Démontrer que $f \equiv (x_i \vee f_{\overline{x_i}}) \wedge (\overline{x_i} \vee f_{x_i})$

On définit la dérivée booléenne par rapport à x_i d'une fonction booléenne
$f : (x_1, x_2, \dots, x_i, \dots, x_n) \longmapsto f(x_1, x_2, \dots, x_i, \dots, x_n)$ par $\dfrac{\partial f}{\partial x_i} = f_{x_i} \oplus f_{\overline{x_i}}$
où le symbole $\oplus$ désigne le ou exclusif `xor`.

c) Démontrer que la valeur de f est indépendante de la valeur de x_i si $\frac{\partial f}{\partial x_i} = 0$ et que la valeur de f dépend de la valeur de x_i si $\frac{\partial f}{\partial x_i} = 1$.

d) Démontrer que $\dfrac{\partial f}{\partial x_i} \equiv \dfrac{\partial \overline{f}}{\partial x_i}$.

Soient f et g deux fonctions booléennes des n variables $x_1, x_2, \dots, x_n$:

e) Démontrer que :

$$\frac{\partial (f \wedge g)}{\partial x_i} \equiv \left(f \wedge \frac{\partial g}{\partial x_i}\right) \oplus \left(g \wedge \frac{\partial f}{\partial x_i}\right) \oplus \left(\frac{\partial f}{\partial x_i} \wedge \frac{\partial g}{\partial x_i}\right)$$

2 Résolution d'une énigme par la logique des propositions

9) Vous êtes perdu sur une piste dans le désert. Vous arrivez à une bifurcation. Chacune des deux pistes est gardée par un sphynx que vous pouvez interroger. Les pistes peuvent soit conduire à une oasis, soit se perdre dans un désert profond (au mieux elle conduisent toutes à une oasis, au pire elles se perdent toutes les deux).

A Le sphynx de droite vous répond : une au moins des deux pistes conduit à une oasis.

B Le sphynx de gauche vous répond : la piste de droite se perd dans le désert.

C Vous savez que les deux sphynx disent tous les deux la vérité ou bien mentent tous les deux.

Soit

OD la proposition “Il y a une oasis au bout de la route de droite”,

OG la proposition “Il y a une oasis au bout de la route de gauche”.

a) Exprimer par une formule de la logique des propositions les affirmations A et B.

b) Exprimer alors la connaissance C.

c) Résoudre l'énigme en utilisant les tables de vérité.

d) Résoudre l'énigme en utilisant les formules de De Morgan.

10) Vous êtes l'ambassadeur de la fédération des planètes unies auprès d'une civilisation extra-terrestre particulièrement susceptible. Il est extrêmement important de respecter leur protocole en ce qui concerne la couleur des vêtements. Pour éviter les incompréhensions, vous disposez d'un assistant de la dernière génération d'intelligence artificielle bio-électronique qui contient toutes les informations disponibles sur leur culture. Mais celui-ci semble perturbé depuis votre arrivée. Il vous donne maintenant des indications sous la forme d'énigmes. Vous avez constaté qu'il vous communiquait systématiquement trois affirmations dont exactement l'une des trois est correcte et les deux autres fausses.

Nous noterons A_1, A_2 et A_3 les propositions associées aux trois affirmations communiquées par votre assistant.

a) Représenter cette règle sous la forme d'une formule du calcul des propositions dépendant de A_1, A_2 et A_3.

Vous devez d'abord rencontrer le gouverneur de la capitale qui est le seul à pouvoir vous obtenir un contact avec le conseil des sages.

Votre assistant vous donne les indications suivantes :

- Porte au moins un vêtement de couleur claire, mais aucun vêtement de couleur sombre !
- Si tu portes au moins un vêtement de couleur claire alors ne porte aucun vêtement de couleur sombre !
- Ne porte aucun vêtement de couleur sombre !

Nous noterons C, respectivement S, les variables propositionnelles correspondant au fait de porter au moins un vêtement de couleur claire, respectivement sombre.

b) Exprimer A_1, A_2 et A_3 sous la forme de formules du calcul des propositions dépendant de C et de S.

c) En utilisant le calcul des propositions (résolution avec des formules de De Morgan), déterminer quelle(s) couleur(s) de vêtement vous devez porter pour rencontrer le gouverneur.

Grâce à ces conseils avisés, vous obtenez un rendez-vous avec le conseil. Vous disposez de vêtements de couleur rouge, verte et bleue. Votre assistant vous transmet ses conseils mais la communication est brouillée juste à la fin et vous ne pouvez plus le contacter avant l'heure du rendez-vous. Voici ce que vous avez réussi à comprendre :

- Ne porte un vêtement de couleur rouge que si tu portes aussi un vêtement de couleur bleue !
- Si tu ne portes pas de vêtement de couleur verte alors ne porte pas de vêtement de couleur bleue !
- Porte au moins un vêtement de couleur bleue ou couleur...!

Nous noterons R, V et B les variables propositionnelles correspondant au fait de porter au moins un vêtement de couleur rouge, verte et bleue.

d) Exprimer A_1, A_2 et A_3 sous la forme de formules du calcul des propositions dépendant de R, V et B.

e) En utilisant le calcul des propositions (résolution avec des tables de vérité), déterminer quelle(s) couleur(s) de vêtement vous devez porter pour votre entrevue avec le conseil des sages.

3 Formes normales

11) Soit $\mathcal{V} = \{v_1, \ldots, v_n\}$ des variables propositionnelles. On appelle littéral toute proposition logique de la forme v_i ou $\neg v_i$. On appelle conjonction (resp. disjonction) de littéraux toute proposition logique de la forme $\ell_1 \wedge \ell_2 \wedge \cdots \wedge \ell_p$ (resp. $\ell_1 \vee \ell_2 \vee \cdots \vee \ell_p$) où $\ell_1, \ldots, \ell_p$ sont des littéraux. On appelle minterme (resp. maxterme) une conjonction (resp. disjonction) de littéraux où chaque variable propositionnelle apparait une fois et une seule.

Soit p une proposition logique.

a) i) Montrer que p est équivalente à une disjonction de mintermes. Montrer de plus que cette disjonction est unique à l'ordre des termes près. *Cela s'appelle la forme normale disjonctive.*

ii) En déduire que p est équivalente à une conjonction de maxtermes. Montrer de plus que cette conjonction est unique à l'ordre des termes près. *Cela s'appelle la forme normale conjonctive.*

b) Déterminer les formes normales disjonctive et conjonctive de

$$(v_1 \wedge v_2) \Leftrightarrow (v_1 \vee v_3)$$

dans le cas où $n = 3$. Expliquer comment modifier les formes obtenues pour quelles soient les formes normales disjonctive et conjonctive dans le cas où $n = 4$.

12) **Calcul d'une forme clausale**

Soit `n` un entier naturel.

On implémente les propositions dont les variables appartiennent à $[\![0, n-1]\!]$ par le type :

```
type proposition =
 | Var of int
 | Non of proposition
 | Ou of proposition * proposition
 | Et of proposition * proposition;;
```

a) En utilisant les règles de De Morgan, écrire une fonction

```
d_neg : proposition -> proposition
```

qui renvoie une proposition sémantiquement égale à la proposition passée en paramètre, mais où les négations ne peuvent avoir pour fils qu'une feuille.

b) Écrire une fonction récursive `d_ou : formule -> formule` qui applique récursivement les propriétés de distributivité $P \vee (Q \wedge R) \equiv (P \vee Q) \wedge (P \vee R)$ ainsi que $(P \wedge Q) \vee R \equiv (P \vee R) \wedge (Q \vee R)$ à chaque fois qu'un nœud `Ou` est père d'un nœud `Et`.

Prouver formellement que votre fonction s'arrête.

c) Une proposition P vérifie la condition "et–ou–non" si, le long de tout chemin de P de la racine à une feuille,

$$\begin{cases} \text{tout nœud “Et” doit se rencontrer avant tout nœud “Ou”} \\ \text{il ne peut y avoir de nœud “Non” qu'éventuellement juste avant la feuille} \end{cases}$$

Écrire une fonction `et_ou_non : formule -> formule` qui renvoie une proposition qui vérifie la condition "et–ou–non" et qui soit sémantiquement égale à la proposition passée en paramètre.

Indication : on pourra appliquer autant de fois que nécessaire la fonction `d_ou` jusqu'à obtenir un point fixe. On pourra admettre que cette fonction termine.

d) Un *littéral* ℓ est une proposition de la forme `Var i` ou `Non (Var i)`, on dit alors que `i` est la variable apparaissant dans ℓ.

Une *clause* est une proposition de la forme $\bigvee_{k=0}^{p-1} \ell_k$ où $\ell_0, \ldots, \ell_{p-1}$ sont des littéraux faisant apparaître des variables 2 à 2 distinctes.

On code cette clause par un tableau `t : int array` de taille n ainsi :

$$\texttt{t.(i)} = \begin{cases} 1 & \text{si le littéral } \texttt{Var i} \text{ est l'un des littéraux } \ell_k \\ -1 & \text{si le littéral } \texttt{Non (Var i)} \text{ est l'un des littéraux } \ell_k \\ 0 & \text{si la variable } \texttt{i} \text{ n'apparaît dans aucun des littéraux } \ell_k \end{cases}$$

Une *forme clausale* est une conjonction de clauses $c_0, \ldots, c_{q-1}$.

On code la forme clausale $\bigwedge_{j=0}^{q-1} c_j$ par le type `(int array) list` où chacun des constituants de la liste est un tableau codant une clause c_j.

Écrire une fonction :

```
fc : proposition -> int -> (int array) list
```

telle que l'appel `fc p n` calcule une forme clausale équivalente à `p`. Le second paramètre est l'entier n fixé en début d'énoncé.