TD 1 : Logique propositionnelle

Ce document est disponible à l'URL `http://www.liafa.jussieu.fr/~sighirea/logver/td_lprop.html`

1 Propriétés

Exercice 1 : On appelle *fonction caractéristique* d'une partie A d'un ensemble E la fonction $\chi_A : E\text{->}\{0,1\}$ définie par :

$$\chi_A(e) = \begin{cases} 1 & si\ e \in A \\ 0 & si\ e \notin A \end{cases}$$

1. Exprimer en termes de fonction caractéristique les opérations d'union, intersection, différence, différence symétrique, complémentation.
2. Si E est un ensemble fini de taille n montrer que la fonction caractéristique d'une partie A de E peut être écrite comme un n-uplet de nombres 0 ou 1 ; en déduire un isomorphisme entre les algèbres de Boole $\mathcal{P}(E)$ et $\{0,1\}^n$.

Exercice 2 : Trouver un polynôme booléen pour la fonction f définie par :

x_1	x_2	x_3	$f(x_1, x_2, x_3)$
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0

Quelle est la table de vérité de la fonction duale? Trouver un polynôme booléen pour la fonction duale.

Exercice 3 : On définit x `nand` $y = \neg(xy)$. Montrer que toutes les opérations booléennes sont exprimables en fonction de `nand`.

2 Formes normale

Rappels :

– Forme normale disjonctive : (*somme de produits*)

$$\phi = \ +_{i=1}^{i=n}(\ \cdot_{j=1}^{j=n_i} [\neg] p_{i_j})$$

– Forme normale conjonctive : (*produits de sommes*)

$$\phi = \ \cdot_{i=1}^{i=n}(\ +_{j=1}^{j=n_i} [\neg] p_{i_j})$$

– Forme normale Reed-Muller : (*xor de produits*)

$$\phi = \ \mathtt{xor}\,_{i=1}^{i=n}(\ .\ _{j=1}^{j=n_i} p_{i_j})$$

Exercice 4 : Mettre en forme normale disjonctive, conjonctive et Reed-Muller les expressions suivantes :

$$\begin{array}{ll} (1) & \neg(p\ .\ \neg(q\ +\ s)) \\ (2) & \neg(p\ .\ (q\ +\ s) \\ (3) & \neg(p\ +\ (q\ .\ \neg s))\ .\ s \end{array}$$

3 Décomposition de Shannon

Soient $x_1, x_2, ..., x_n$ un ensemble de variables booléennes et f une expression booléenne de ces variables ($f : \mathbb{B}^n$ -> $\mathbb{B}$).
Définition : La *décomposition de Shannon* d'une fonction f selon la variable x_k est le couple (unique) de formules :

$$f_{\neg x_k} = f[faux/x_k] \quad , \quad f_{x_k} = f[vrai/x_k]$$

On a $f = (\neg x_k\ .\ f_{\neg x_k})\ +\ (x_k\ .\ f_{x_k})$.
Définition : L'*arbre de Shannon* pour un ordre fixé des variables $x_1, x_2, ..., x_n$ est obtenu par la décomposition itérative de f selon les variables $x_1, x_2, ..., x_n$. L'*arbre réduit* de Shannon est obtenu par élimination des sommets dont les deux sous-arbres sont égaux.

Exercice 5 : Ecrire l'arbre de Shannon pour la formule

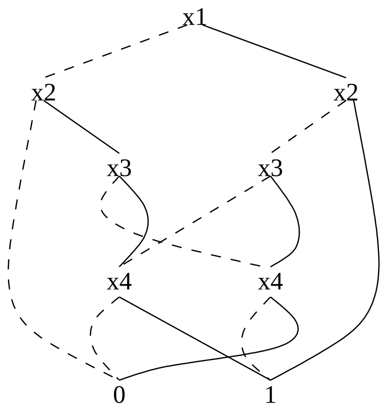
$$f(x_1, x_2, x_3, x_4) = (x_1\ .\ (x_3\ \mathtt{xor}\ x_4))\ +\ (x_2\ .\ (x_3 \mathtt{<=>} x_4))$$

pour les ordres suivants des variables :

- $x_1 < x_2 < x_3 < x_4$
- $x_3 < x_4 < x_1 < x_2$

4 Graphes binaires de décision (BDD)

Définition : Un BDD est un graphe obtenu à partir de arbre réduit de Shannon par partage des sous-arbres identiques.
Exemple : Le BDD de la formule $(x_1\ .\ (x_3\ \mathtt{xor}\ x_4))\ +\ (x_2\ .\ (x_3 \mathtt{<=>} x_4))$ pour l'ordre $x_1 < x_2 < x_3 < x_4$ est :



Exercice 6 : Ecrire le BDD de la formule ci-dessus pour l'ordre $x_3 < x_4 < x_1 < x_2$