Algebre de boole exercice corrigé

Algèbre de boole exercice corrigé. Exercice corrigé sur algebre de boole.

Une page de Wikiversité, la communauté pédagogique libre. En raison de limitations techniques, la typographie souhaitable du titre, « Exercice : Algèbre de BooleLogique de base/Exercices/Algèbre de Boole », n'a pu être restituée correctement ci-dessus. Réduire les équations en utilisant les propriétés de l'algèbre de Boole: À 2 variables[modifier | modifier le wikicode] Équation 1[modifier | modifier le wikicode] A = a + a · b {\displaystyle A=a+a\cdot b} Équation 2[modifier | modifier le wikicode] B = a · (a + b) {\displaystyle B=a\cdot (a+b)} Équation 3[modifier | modifier le wikicode] C = a + (a ¯ · b) {\displaystyle C=a+({\bar {a}}\cdot b)} Équation 4[modifier | modifier le wikicode] D = (a + b) · (a + b ¯) {\displaystyle D=(a+b)\cdot (a+{\bar {b}})} Équation 5[modifier | modifier le wikicode] G = (a ¯ + b ¯) + (a ¯ + b) + (a + b ¯) {\displaystyle G=({\bar {a}}+{\bar {b}})+({\bar {a}}+b)+(a+{\bar {b}})} Équation 6[modifier | modifier le wikicode] H = (a ¯ + b) + (a + b ¯) + (a + b) {\displaystyle H=({\bar {a}}+b)+(a+{\bar {b}})+(a+b)} À 3 variables[modifier | modifier le wikicode] Équation 1[modifier | modifier le wikicode] E = (a + b) · (a + c) {\displaystyle E=(a+b)\cdot (a+c)} Équation 2[modifier | modifier le wikicode] F = (a + b) · (a ¯ + c) {\displaystyle F=(a+b)\cdot ({\bar {a}}+c)} Équation 3[modifier | modifier le wikicode] P = (a + b ¯) · (b + c ¯) · (c + a ¯) {\displaystyle P=(a+{\bar {b}})\cdot (b+{\bar {c}})\cdot (c+{\bar {a}})} Équation 4[modifier | modifier le wikicode] I = (b ¯ + a ¯) · (a · c + b ¯) {\displaystyle I=({\bar {b}}+{\bar {a}})\cdot (a\cdot c+{\bar {b}})} Équation 5[modifier | modifier le wikicode] Q = (a + b + c) · (a + b ¯ + c) · (a + b ¯ + c ¯) {\displaystyle Q=(a+b+c)\cdot (a+{\bar {b}}+c)\cdot (a+{\bar {b}}+{\bar {c}})} Équation 6[modifier | modifier le wikicode] T = a · b · c + a · b ¯ · c + a · b · c ¯ {\displaystyle T=a\cdot b\cdot c+a\cdot {\bar {b}}\cdot c+a\cdot b\cdot {\bar {c}}} Équation 7[modifier | modifier le wikicode] V = (a + b) · (a + c) + (b + c) · (b + a) + (c + a) · (c + b) {\displaystyle V=(a+b)\cdot (a+c)+(b+c)\cdot (b+a)+(c+a)\cdot (c+b)} Équation 8[modifier | modifier le wikicode] K = a · c · (a ¯ + b + c ¯) {\displaystyle K=a\cdot c\cdot ({\bar {a}}+b+{\bar {c}})} Équation 9[modifier | modifier le wikicode] L = a b + a ¯ b + a c + a ¯ c {\displaystyle L=ab+{\bar {a}}b+ac+{\bar {a}}c} À 4 variables[modifier | modifier le wikicode] Équation 1[modifier | modifier le wikicode] U = (a ¯ + b) · (a + b + d) · d ¯ {\displaystyle U=({\bar {a}}+b)\cdot (a+b+d)\cdot {\bar {d}}} Équation 2[modifier | modifier le wikicode] Z = (a · b + c + d) · a · b {\displaystyle Z=(a\cdot b+c+d)\cdot a\cdot b} Équation 3[modifier | modifier le wikicode] J = c · (b + c) + (a + d) · (a ¯ + d) · c ¯ {\displaystyle J=c\cdot (b+c)+(a+d)\cdot ({\bar {a}}+d)\cdot {\bar {c}}} Logique de base Sommaire Tableau de Karnaugh 1   Architecture des ordinateursCorrigé du TD 3 : Algèbre de Boole Arnaud Giersch, Benoît Meister et Frédéric Vivien 1. Montrer comment l'opérateur et peut être obtenu à partir des opérateurs ou et non . De même pour l'opérateur ou avec les opérateurs et et non . Correction : non(a ou b) = (non a) et (non b) ⇒ non((non a) ou (non b)) = a et bnon(a et b) = (non a) ou (non b) ⇒ non((non a) et (non b)) = a ou b 2. On note respectivement les opérateurs ou , et , xor et non par + , · , ⊕ et . Montrer à l'aide de tables de véritéque A ⊕ B = A · B + A · B et que A ⊕ B = (A + B) · (A + B) Correction : Tables de vérités : A B A B A ⊕ B A · B A · B A · B + A · B 1 1 0 0 0 0 0 0 01 0 0 1 1 0 1 10 1 1 0 1 1 0 10 0 1 1 0 0 0 0 A B A B A ⊕ B A + B A + B (A + B) · (A + B) 1 1 0 0 0 1 0 01 0 0 1 1 1 1 10 1 1 0 1 1 1 10 0 1 1 0 0 1 03. Montrer que A +(A · B) = A + B et que A · (A + B) = A · B Correction : On utilise la distributivité de l'opérateur   ou sur l'opérateur   et , et inversement : A +(A · B) = (A + A) . (A + B) = 1 . (A + B) = A + B A · (A + B) = (A · A)+(A · B) = 0 +(A · B) = A · B 4. Déterminer le complément de l'expression A + B · C Correction : On utilise les lois de de Morgan; l'opérateur   et est prioritaire : A + B · C   = A · B · C   = A · (B + C) = A · B + A · C 5. Montrer que les deux règles d'associativité sont duales, i.e. montrer qu'à partir de la règle d'associativité del'opérateur ou , on peut déduire, en utilisant les lois de de Morgan, l'associativité de l'opérateur et (et inverse-ment). Correction : A +(B + C) = (A + B)+ C   ⇔ A +(B + C) = (A + B)+ C   ⇔ A · (B · C) = (A · B) · C A, B, et C sont des variables muettes. Par changement de variable { (A → A), (B → B), (C →) C } on obtient la propriété d'associativité du ou : A · (B · C) = (A · B) · C 1 6. Écrire l'expression A ⊕ B uniquement avec les opérateurs ou , et et nonCorrection : D'après 2. : A ⊕ B = A · B + A · B ⇔ A ⊕ B = A · B + A · B ⇔ A ⊕ B = (A + B) · (A + B) 7. Montrer que la fonction nor forme un groupe logique complet. Correction : Pour cela, on montre que la fonction nor permet d'exprimer tous les opérateurs logiques : – non : nor (A , A) = A – et : nor (nor (A , A) , nor (B , B)) = nor (A , B) = A + B = A · B – ou : nor (nor (A , B) , nor (A , B)) = nor (A , B) = (A + B) = (A + B) . 8. Simplifier au maximum les expressions logiques suivantes.(a) A · B + A · B Correction : A · B + A · B = (A + A) · B = 1 · B = B (b) (A + B) · (A + B) Correction : (A + B) · (A + B) = A + B · B = A + 0 = A (c) A + A · B Correction : A + A · B = A · 1 + A · B = A · (1 + B) = A · 1 = A (d) A · (A + B) Correction : A · (A + B) = (A + 0) · (A + B) = A + 0 · B = A + 0 = A (e) A · B + A + B + C + D Correction : A · B + A + B + C + D = (A + B) · (A + B + C + D)= (A + B) · ((A + B)+(C + D)) donc, d'après l'exercice 8d, = A + B (f) A + B · C + A · (B · C) · (A · D + B) Correction : A + B · C + A · (B · C) · (A · D + B) = (A + B · C)+(A + B · C) · (A · D + B) d'après l'exercice 3, A + B · C + A · (B · C) · (A · D + B) = (A + B · C)+(A · D + B) = (A + A · D)+(B + B · C) d'après l'exercice 8c, A + B · C + A · (B · C) · (A · D + B) = A + B (g) (A ⊕ B) · B + A · B 2   Correction : d'après l'exercice 2, (A ⊕ B) · B + A · B = (A · B + A · B) · B + A · B = A · B + A · B · B + A · B = A · B + A · Bd'après l'exercice 8a, = B (h) A + A · B + A · B Correction : A + A · B + A · B = (A + A · B)+ A · Bd'après l'excercice 3, A + A · B + A · B = (A + B)+(A + B) = 19. Démontrer que toute fonction à trois variables F (A , B , C) est égale à F (A , B , C) = A · F (1 , B , C)+ A · F (0 , B , C) Correction : A est une variable booléenne : les deux valeurs qu'elle peut prendre sont   0 et   1 : – si A = 0 , 0 · F (1 , B , C)+ 1 · F (0 , B , C) = F (0 , B , C) = F (A , B , C) ; – si A = 1 , 1 · F (1 , B , C)+ 0 · F (0 , B , C) = F (1 , B , C) = F (A , B , C) . 10. Montrer que les lois de de Morgan s'étendent à un nombre quelconque de variables. Correction : (a) A 1 · A 2 ···· A n = A 1 + A 2 + ··· + A n avec n ≥ 2 . La démonstration se fait par récurrence sur n (le nombrede variables).n = 2 c'est la loi de de Morgan « basique »;n > 2 on utilise l'associativité de + et · : A 1 · A 2 ···· A n = (A 1 · A 2 ···· A n − 1) · A n = (A 1 · A 2 ···· A n − 1)+ A n = (A 1 + A 2 + ··· + A n − 1)+ A n = A 1 + A 2 + ··· + A n − 1 + A n (b) A 1 + A 2 + ··· + A n = A 1 · A 2 ···· A n avec n ≥ 2 . Le raisonnement est similaire. 11. Génération et simplification d'expressions logiques Considérer la fonction définie par la table de vérité ci-dessous : A B C F (A , B , C) 0 0 0 0 0 0 1 10 1 0 10 1 1 01 0 0 11 0 1 11 1 0 11 1 1 03 Tableaux de Karnaugh Simplification d'expressions sans tableau Expression avec des NAND et des NOR Tableaux de Karnaugh Simplifier les tableaux de Karnaugh suivants : Simplification d'expressions sans tableau Haut de page L'exercice consiste à simplifier les expressions suivantes sans utiliser de tableau de Karnaugh : Expression avec des NAND et des NOR Haut de page L'exercice consiste à exprimer uniquement avec des NAND puis uniquement avec des NOR l'expression définie par le tableau suivant : Retour au sommaire des exercicesRemonter en haut de la page