5. Montrer que les deux règles d’associativité sont duales, i.e. montrer qu’à partir de la règle d’associativité de l’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*j)*,* (*B* → *B*j)*,* (*C* →)*C*j} *on obtient la propriété d’associativité du* **ou** *: A*j · (*B*j · *C*j) = (*A*j · *B*j) · *C*j

Algèbre de Boole

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 b non(a et b) = (non a) ou (non b)* ⇒ *non((non a) et (non b)) = a ou b*

**Correction :** *Tables de vérités :*

1. On note respectivement les opérateurs **ou**, **et**, **xor** et **non** par +*,* ·*,* ⊕ et . Montrer à l’aide de tables de vérité que *A* ⊕ *B* = *A* · *B* + *A* · *B* et que *A* ⊕ *B* = (*A* + *B*) · (*A* + *B*)



0

1

1

0

0

0

1

0

0

1

0

0

0

1

1

0

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| *A* | *B* | *A* | *B* | *A* ⊕ *B* | *A* · *B* | *A* · *B* | *A* · *B* + *A* · *B* |
| 1  1  0  0 | 1  0  1  0 | 0  0  1  1 | 0  1  0  1 |  |  |  |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| *A* | *B* | *A* | *B* | *A* ⊕ *B* | *A* + *B* | *A* + *B* | (*A* + *B*) · (*A* + *B*) |
| 1  1  0  0 | 1  0  1  0 | 0  0  1  1 | 0  1  0  1 |  |  |  |  |

1. Montrer que *A* + (*A* · *B*) = *A* + *B* et que *A* · (*A* + *B*) = *A* · *B*

0

1

1

0

1

1

1

0

0

1

1

1

0

1

1

0

**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*

**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*

https://cdn.discordapp.com/attachments/494837642437984256/515201450377216011/pour\_toi.jpg



1. Écrire l’expression *A* ⊕ *B* uniquement avec les opérateurs **ou**, **et** et **non**

**Correction :** *D’après 2. :*

*A* ⊕ *B* = *A* · *B* + *A* · *B* ⇔ *A* ⊕ *B* = *A* · *B* + *A* · *B* ⇔ *A* ⊕ *B* = (*A* + *B*) · (*A* + *B*)

Simplifier au maximum les expressions logiques suivantes.

8.

* 1. *A* · *B* + *A* · *B*

**Correction :**

*A* · *B* + *A* · *B* = (*A* + *A*) · *B* = 1 · *B* =

*B*

**Correction :**

(*A* + *B*) · (*A* + *B*) = *A* + *B* · *B* = *A* + 0 = *A*

1. *A* · (*A* + *B*)

**Correction :**

*A* + *A* · *B* = *A* · 1 + *A* · *B* = *A* · (1 + *B*) = *A* · 1 = *A*

**Correction :**

*A* · (*A* + *B*) = (*A* + 0) · (*A* + *B*) = *A* + 0 · *B* = *A* + 0 = *A*

**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*

1. *A* · *B* + *A* + *B* + *C* + *D*

(f) *A* + *B* · *C* + *A* · (*B* · *C*) · (*A* · *D* + *B*)

# 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 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 |

* 1. Générer une expression logique correspondante :
     1. sous forme de sommes de produits ;

**Correction :**

*A* · *B* · *C* + *A* · *B* · *C* + *A* · *B* · *C* + *A* · *B* · *C* + *A* · *B* ·

*C*

* + 1. sous forme de produits de s
    2. ommes.
  1. Simplifier les deux expressions en utilisant les règles de l’algèbre de Boole.

**Correction :**

*A* · *B* · *C* + *A* · *B* · *C* + *A* · *B* · *C* = (*A* + *B* + *C*) · (*A* + *B* + *C*) · (*A* + *B* + *C*)

*i.*

*ii.*

(*A* + *B* + *C*) · (*A* + *B* + *C*) · (*A* + *B* + *C*)

= (*A* · *A* + *A* · *B* + *A* · *C* + *B* · *A* + *B* · *B* + *B* · *C* + *C* · *A* + *C* · *B* + *C* · *C*) · (*A* + *B* + *C*)

= (*A* + *A* · *B* + *A* · *C* + *A* · *B* + *A* · *C* + *B* · *C* + *B* · *C*) · (*A* + *B* + *C*)

= *A* · *A* + *A* · *B* · *A* + *A* · *C* · *A* + *A* · *B* · *A* + *A* · *C* · *A* + *B* · *C* · *A* + *B* · *C* · *A* + *A* · *B* + *A* · *B* · *B* + *A* · *C* · *B* + *A* · *B* · *B* + *A* · *C* · *B* + *B* · *C* · *B* + *B* · *C* · *B* + *A* · *C* + *A* · *B* · *C* + *A* · *C* · *C* + *A* · *B* · *C* + *A* · *C* · *C* + *B* · *C* · *C* + *B* · *C* · *C*

= *A* · *B* · *C* + *A* · *B* + *A* · *B* · *C* + *A* · *B* · *C* + *A* · *C* + *B* · *C* + *B* · *C*

= (*A* · *B*) · (1 + *C* + *C*) + *B* · *C* + (*A* + 1) · (*B* · *C*)

= *A* · *B* + *B* · *C* + *B* · *C*

= *A* · *B* + (*B* ⊕ *C*)

Considérer les fonctions logiques suivantes. Pour chacune d’elles,

12.

* construire le diagramme de Karnaugh ;
* utiliser le diagramme pour simplifier les expressions.

1. *F*1(*A, B,C*) = *A* · *B* · *C* + *A* · *B* · *C* + *A* · *B* · *C*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *BC*  *A* | *BC* | *BC* | *BC* | *BC* |
| *A* |  |  |  |  |
| *A* |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *BC*  *A* | *BC* | *BC* | *BC* | *BC* |
| *A* |  |  |  |  |
| *A* |  |  |  |  |

**Correction :** *La table de Karnaugh est présentée figure 1.*

FIG. 1 – Table de Karnaugh pour *F*1(*A, B,C*).

FIG. 2 – Table de Karnaugh pour *F*2(*A, B,C*).

1. *F*2(*A, B,C*) = *A* · *B* · *C* + *A* · *B* + *A* · *B* · *C*

**Correction :** *La table de Karnaugh est présentée figure 2. Expression simplifiée : F*2(*A, B,C*) = *A* · *C* + *B* · *C*

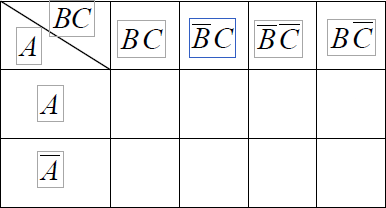
**Correction :**

3(*A, B,C*) = *A* · *B* + *A* · *B* · *C* + *B* · *C* + *A* · *B* · *C*

= *A* · *B* · *C* + *A* · *B* · *C* + *A* · *B* · *C* + *A* · *B* · *C* + *A* · *B* · *C* + *A* · *B* · *C*

1. *F*3(*A, B,C*) = *A* · *B* + *A* · *B* · *C* + *B* · *C* + *A* · *B* · *C*

*La table de Karnaugh est présentée figure 3. Expression simplifiée : F*3(*A, B,C*) = *B* + *A* · *C*

1. *F*4(*A, B,C, D*) = *B* · *C* · *D* + *A* · *B* · *D* + *A* · *B* · *C* · *D*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *CD*  *AB* | *C D* | *C D* | *C D* | *C D* |
| *A B* |  |  |  |  |
| *A B* |  |  |  |  |
| *A B* |  |  |  |  |
| *A B* |  |  |  |  |

FIG. 3 – Table de Karnaugh pour *F*3(*A, B,C*).

FIG. 4 – Table de Karnaugh pour *F*4(*A, B,C, D*).

1. *F*5(*A, B,C, D*) = *A* + *A* · *B* + *A* · *B* · *C* + *A* · *B* · *C* · *D*

**Correction :**

*F*5(*A, B,C, D*) = *A* · *B* · *C* · *D* · + · *A* · *B* · *C* · *D* + *A* · *B* · *C* · *D* + *A* · *B* · *C* · *D* + *A* · *B* · *C* · *D*

+ *A* · *B* · *C* · *D* + *A* · *B* · *C* · *D* + *A* · *B* · *C* · *D* + *A* · *B* · *C* · *D* + *A* · *B* · *C* ·

*D*

+ *A* · *B* · *C* · *D* + *A* · *B* · *C* · *D* + *A* · *B* · *C* · *D* + *A* · *B* · *C* · *D* + *A* · *B* · *C* · *D*

*table de Karnaugh est présentée figure 5. Expression simplifiée : F*5(*A, B,C, D*) = *B* + *A* + *C*

FIG. 5 – Table de Karnaugh pour *F*5(*A, B,C, D*).

FIG. 6 – Table de Karnaugh pour *F*6(*A, B,C, D*).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| *CD*  *AB* | *C D* | *C D* | *C D* | *C D* |
| *A B* |  |  |  |  |
| *A B* |  |  |  |  |
| *A B* |  |  |  |  |
| *A B* |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *CD*  *AB* | *C D* | *C D* | *C D* | *C D* |
| *A B* |  |  |  |  |
| *A B* |  |  |  |  |
| *A B* |  |  |  |  |
| *A B* |  |  |  |  |

1. *F*6(*A, B,C, D*) = *A* · *B* · *D* + *A* · *C* · *D* + *A* · *B* · *C* · *D* + *A* · *B* · *D* + *B* · *C* · *D* + *A* · *B* · *C* · *D*