Introduction, notion de booléen

I Valeurs booléennes et opérateurs booléens fondamentaux

1) Booléens

Exercice 1:
On donne les cinq booléens A, B, C, D, E suivants :
A = 1
B = (3 > 5)
C = ("x" est présent dans "OxO")
D = 0
E = (B == 0)

1. Quelle est la nature de la variable A : une chaîne de caractères ? un nombre ? un bit ? un booléen ? Qu'est-ce qui vous permet de le savoir ?
2. Donner la valeur des cinq booléens.

Exercice 2 : Sauriez-vous trouver les valeurs de trois booléens A, B et C tels que les conditions suivantes soient vérifiées toutes les deux à la fois ?
(A ET B) = 0
(B ET C) = 1

2) Opérateurs booléens fondamentaux : and, or, not.

Exercice 3 :
Soit X un nombre entier. On considère, construits à partir des affirmations (ou plutôt *expressions booléennes*) suivantes, trois booléens A, B et C.
A = (X < 10),
B = (X > 3),
C = (X est un nombre pair).
Compléter le tableau suivant en indiquant dans chacune des cases la valeur booléenne des expressions indiquées.

X	A	B	C	not A	A and B	A or C
5	1	1	0	0		
8						
11						
12						
3						
2						
4355						

II Expressions Booléennes

1) Propriétés des opérateurs

- Associativité : A and B and C = (A and B) and C = A and (B and C)
  A or B or C = (A or B) or C = A or (B or C)
- Commutativité : A and B = B and A
  A or B = B or A
- Distributivité : A and (B or C) = (A and B) or (A and C)
- Priorité : A and B or C = (A and B) or C
  A or B and C = A or (B and C)
- Priorité de not : not A and B = (not A) and B
  not A or B = (not A) or B

Licence Creative Commons CC – BY – SA , C. Lejeune et J. de Villèle

Exercice 4 :
Soit X un caractère alphabétique. On considère les trois booléens suivants A, B et C obtenus à partir de trois expressions booléennes (ou "affirmations") :
A = (X est présent dans la chaîne de caractères "Claude Shannon et Georges Boole"),
B = (X correspond à une lettre voyelle),
C = (X correspond à une lettre majuscule).
1) Compléter le tableau suivant en indiquant dans chacune des cases la valeur booléenne des expressions indiquées. On commencera par la première ligne, puis par la seconde etc.

X	A	B	C	not A	A and B and C	A or B or C	B and C	A or B	A or B and C	not A and B
"g"	1	0	0	0						
"A"										
"a"										
"B"										
"b"										
"m"										
"Z"										

2) Si il y en a eu, quelles sont les colonnes qui ont été délicates à remplir ? Pourquoi ?

3) Justifier brièvement que, quel que soit le choix de X dans tout l'alphabet minuscule ou majuscule, on a :
(A and B and C) = 0

2) Dresser la table de vérité d'une expression booléenne

Exercice 5
Soit A, B, C trois variables booléennes. Dresser la table de vérité de :
not A and B and C

A	B	C		
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

Exercice 6
Soit A, B, C trois variables booléennes. Dresser la table de vérité de :
A and B or A and C

A	B	C			
0	0				
0	0				
0	1				
0	1				
1	0				
1	0				
1	1				
1	1				

Licence Creative Commons CC – BY – SA , C. Lejeune et J. de Villèle

Exercice 7

Soit A, B, C trois variables booléennes. Dresser la table de vérité de :
A and (B or C)
Comparer avec l'exemple précédent.

A	B	C		

3) Le cas du XOR

L'opérateur booléen xor est le e**X**clusive **OR**, en français le ou exclusif. On peut le comprendre de deux façons différentes (qui sont équivalentes).
On peut ainsi voir que A xor B est vrai lorsqu'un seul - exclusivement un seul - des deux booléens A ou B est vrai (A xor B est faux sinon).
On peut aussi voir que A xor B est vrai lorsque A et B sont différents (A xor B est faux sinon).
Pour lever toute ambiguité, sa table de vérité est donnée ci-contre.

A	B	A xor B
0	0	0
0	1	1
1	0	1
1	1	**0**

Exercice 8 :
Soit A et B deux variables booléennes. On définit alors les expressions booléennes suivantes :
C = (A and not B) or (not A and B)
D = not ((A and B) or (not A and not B))
E = ((not A) or (not B)) and (A or B)
Choisir une de ces trois expressions booléennes au hasard et en dresser la table de vérité.
Vérifier qu'elle correspond à la table de vérité de A xor B.

4) Exercice 9 : les lois de De Morgan

Exercice 9
Les lois de De Morgan sont des identités entre propositions logiques. Elles ont été formulées par le mathématicien britannique Augustus De Morgan (1806-1871).
Elles énoncent que, quelles que soient les valeurs des variables booléennes A et B on a :

not (A and B) = (not A) or (not B)

ainsi que :

not (A or B) = (not A) and (not B)

Question 1 :
Quelles sont les parenthèses que l'on aurait pu enlever dans les deux égalités ci-dessus ?

Question 2 :
Dresser la table de vérité de not (A and B) puis dresser la table de vérité de (not A) or (not B).
Que constate-t-on ? Qu'est ce que cela démontre ?

Question 3 :
Faire de même pour la seconde loi de De Morgan.

Licence Creative Commons CC – BY – SA , C. Lejeune et J. de Villèle

III Applications : addition binaire et circuits combinatoires

1) Addition Binaire : Exemples

Ceux qui ont oublié les conversions de base 10 vers base 2 iront utilement relire leur cours.

Exercice 10 :

a) Déterminer l'écriture de 183 en base 2

b) Soit $n = 01101011_2$ un entier représenté en base 2. Déterminer l'écriture de n en base 10.

Exercice 11 :

Calculer les additions suivantes (en base 10) :

	4	5	1	8	3
+	2	4	1	4	5

	7	5	8	7	4
+	6	7	5	9	8

	2	2	2	2	2
+	8	7	6	5	4

	2	3	4	5	6
+	7	8	9	1	0

Exercice 12 :

Calculer les additions suivantes (en base 2) :

	1	0	1	1	1
+	0	0	1	0	1

	1	0	1	0	1
+	0	1	0	1	0

	1	0	1	0	1
+	0	1	0	1	1

	1	1	1	1	0
+	1	1	0	1	0

Licence Creative Commons CC – BY – SA , C. Lejeune et J. de Villèle

2) Addition Binaire : formalisation grâce à une table de vérité

(r_n) correpond à la ligne des retenues, (a_n) à la ligne du nombre A, (b_n) à la ligne du nombre B et (s_n) à la ligne de la somme de A et B.

$$\begin{array}{cccccc} \dots & r_{n+1} & r_n & \dots & r_1 & r_0 \\ \dots & a_{n+1} & a_n & \dots & a_1 & a_0 \\ \dots & b_{n+1} & b_n & \dots & b_1 & b_0 \\ \hline \dots & s_{n+1} & s_n & \dots & s_1 & s_0 \end{array}$$

$$\begin{array}{cccccc} 1 & 1 & 1 & 1 & & \\ & 1 & 0 & 1 & 1 & 1 \\ + & 0 & 1 & 0 & 1 & 0 \\ \hline 1 & 0 & 0 & 0 & 0 & 1 \end{array}$$

Exercice de cours 13 :

1) Compléter la table de vérité ci-contre :

r_n	a_n	b_n	s_n	r_{n+1}
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

2) Soit $nb\_vrai(a, b, r)$ la fonction qui, à tout triplet de booléens (a, b, r), associe le nombre de 1 parmi a, b et r. Par exemple $nb\_vrai(1, 0, 1) = 2$ alors que $nb\_vrai(1, 1, 1) = 3$.

Donner les valeurs de s_n et $r_{(n+1)}$ lorsque :
. $nb\_vrai(a_n, b_n, r_n) = 0$ alors $s_n =$ et $r_{n+1} =$
.$nb\_vrai(a_n, b_n, r_n) = 1$ alors $s_n =$ et $r_{n+1} =$
. $nb\_vrai(a_n, b_n, r_n) = 2$ alors $s_n =$ et $r_{n+1} =$
. $nb\_vrai(a_n, b_n, r_n) = 3$ alors $s_n =$ et $r_{n+1} =$

3) Compléter l'expression booléenne suivante :

$$s_n = (nb\_vrai(a_n, b_n, r_n) = \cdots)\ ou\ (nb\_vrai(a_n, b_n, r_n) = \cdots)$$

4) Compléter l'expression booléenne suivante :

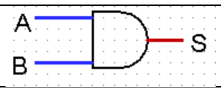
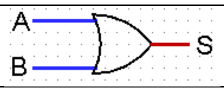
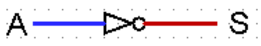
$$r_{n+1} = (nb\_vrai(a_n, b_n, r_n) = \cdots)\ ou\ (nb\_vrai(a_n, b_n, r_n) = \cdots)$$

5) Relier les expressions booléennes qui sont équivalentes entre elles par des traits :

$nb\_vrai(a, b, r) = 0$ ■	■ $a\ and\ b\ and\ c$
$nb\_vrai(a, b, r) = 1$ ■	■ $(a\ and\ not\ b\ and\ not\ c)$ $or\ (not\ a\ and\ b\ and\ not\ c)$ $or\ (not\ a\ and\ not\ b\ and\ c)$
$nb\_vrai(a, b, r) = 2$ ■	■ $not\ a\ and\ not\ b\ and\ not\ c$
$nb\_vrai(a, b, r) = 3$ ■	■ $(a\ and\ b\ and\ not\ c)$ $or\ (not\ a\ and\ b\ and\ c)$ $or\ (a\ and\ not\ b\ and\ c)$

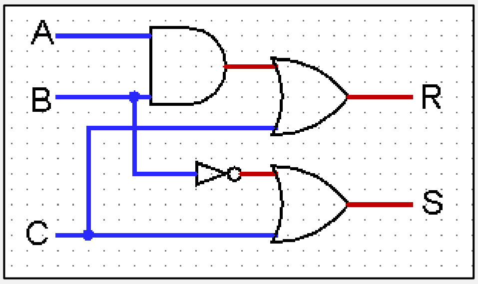
Licence Creative Commons CC – BY – SA , C. Lejeune et J. de Villèle

3) Circuits combinatoires

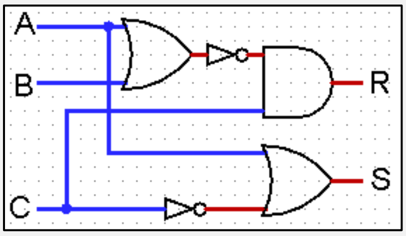
$S = A\ and\ B$	$S = A\ or\ B$	$S = not\ A$
		

<u>Exercice de cours 14 :</u>
En utilisant le circuit combinatoire ci-contre :
1) Déterminer l'expression booléenne de R en fonction de A, B et C.

2) Déterminer l'expression booléenne de S en fonction de A, B et C.



<u>Exercice 15 :</u>
En utilisant le circuit combinatoire ci-contre :
1) Déterminer l'expression booléenne de R en fonction de A, B et C.
2) Déterminer l'expression booléenne de S en fonction de A, B et C.



<u>Exercice 16 :</u>
En effectuant éventuellement une recherche sur internet, trouver le schéma d'un circuit combinatoire réalisant l'expression booléenne de s_n en fonction de a_n, b_n et r_n où les notations s_n, a_n, b_n et r_n correspondent aux notations de l'addition binaire (III.2))

Licence Creative Commons CC – BY – SA , C. Lejeune et J. de Villèle