Álgebra de Boole

Introducción

El álgebra de Boole es una de las teorías básicas que sentaron los precedentes que permitieron construir las computadoras. Partiremos de un conjunto de axiomas para luego ver algunas de las propiedades que se verifican en esta álgebra.

Axiomas

1. Existe un conjunto G de objetos, sujetos a una relación de equivalencia, denotada por "=" que satisface el principio de sustitución. Esto significa que si $a = b$, b puede sustituir a a en cualquier expresión que la contenga, sin alterar la validez de la expresión.

2. Operaciones:
   a) Se define una regla de combinación "+", de forma que $a+b$ está en G siempre que al menos a o b lo estén.
   b) Se define una regla de combinación "·", de forma que $a \cdot b$ está en G siempre que al menos a o b lo estén.

3. Neutros:
   a) Existe un elemento 0 en $G / \forall a \in G, a+0=a$
   b) Existe un elemento 1 en $G / \forall a \in G, a \cdot 1 = a$

4. Conmutativos:
   Para todo par de elementos a y b pertenecientes a G se cumple:
   a) $a+b=b+a$
   b) $a \cdot b = b \cdot a$

5. Distributivos:
   Para toda terna de elementos a,b,c pertenecientes a G se cumple:
   a) $a+(b \cdot c)=(a+b) \cdot (a+c)$
   b) $a \cdot (b+c) = (a \cdot b)+(a \cdot c)$

6. Complemento:
   Para cada elemento a de G, existe un elemento $\overline{a}$ tal que:
   a) $a \cdot \overline{a} = 0$
   b) $a + \overline{a} = 1$

7. Existen por lo menos dos elementos x, y en G tal que $x \neq y$

Modelo aritmético: suma y producto

El ejemplo más simple de álgebra de Boole se compone de un conjunto G de dos elementos: 0 y 1; y ésta es el álgebra de Boole que usaremos.

Se puede demostrar que la única forme de definir las reglas de combinación de forma de satisfacer los axiomas es la siguiente:

$$
\begin{array}{ll}
0+0=0 & 0\cdot 0=0 \\
0+1=1 & 0\cdot 1=0 \\
1+0=1 & 1\cdot 0=0 \\
1+1=1 & 1\cdot 1=1
\end{array}
$$

Propiedades

Dualidad

Cada propiedad tiene una dual (que también es cierta) y que se obtiene cambiando cada 0 por 1 y cada $+$ por $\cdot$ (y viceversa)

Asociativas

$$a+(b+c)=(a+b)+c$$
$$a\cdot(b\cdot c)=(a\cdot b)\cdot c$$

Idempotencia

$$a+a=a$$
$$a\cdot a=a$$

Neutros cruzados

$$a+1=1$$
$$a\cdot 0=0$$

Involución

$$\overline{\overline{a}}=a$$

De Morgan

$$\overline{(a+b)}=\overline{a}\cdot\overline{b}$$
$$\overline{(a\cdot b)}=\overline{a}+\overline{b}$$

Modelo lógico

Podemos representar mediante un modelo lógico equivalente al álgebra de Boole, si realizamos la siguiente correspondencia:

0 será F (falso)
1 será V (verdadero)
$+$ será $\vee$ (OR, disyunción)
$\cdot$ será $\wedge$ (AND, conjunción)
$\bar{}$ será $\neg$ (NOT, negación)

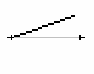
Se puede ver fácilmente que con esta forma de representar el álgebra de Boole se respetan las definiciones clásicas de las tablas de verdad de la lógica proposicional.

Debido a este isomorfismo, tenemos entonces que podemos asignar un semántica lógica al álgebra de Boole y a sus expresiones; o viéndolo al revés, disponemos de un modelo aritmético (binario) para realizar aritmética con las proposiciones lógicas.

Modelo circuital

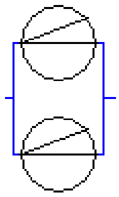
Otra forma alternativa de ver el álgebra de Boole es por medio de circuitos, donde se tienen llaves abiertas por dónde no pasa corriente, y laves cerradas por dónde sí pasa corriente (es recomendable que el lector entienda que esta forma de entender la semántica de abierto y cerrado es sólo aplicable a la electricidad y en el resto de las circunstancias de su vida prefiera pasar por los lugares abiertos que por los cerrados).

Este es un ejemplo de cómo podrían verse las llaves eléctricas que mencionábamos:

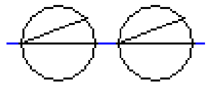
Llave abierta (no pasa corriente) 	Llave cerrada (pasa corriente)

Con respecto a los operadores $+$ y $\cdot$, podemos representarlos poniendo respectivamente dos llaves en paralelo o en serie.

En paralelo:



En serie:



Esta forma de representar el álgebra de Boole es fundamental para la construcción de las computadoras.

Funciones booleanas y conectivas

Trabajaremos en general con los siguientes tipos de funciones booleanas:

- Unarias, $\{f : G \to G\}$
- Binarias, $\{f : G \times G \to G\}$

Como sabemos al ser $|G| = 2$ tenemos que hay $2^2 = 4$ funciones unarias posibles, y son estas:

x	$f(x)$
0	0
1	0

x	$f(x)$
0	0
1	1

x	$f(x)$
0	1
1	0

x	$f(x)$
0	1
1	1

De todas estas, la que tendrá más interés para nosotros es la tercera, a la que que llamaremos NOT.

De la misma forma, tenemos que existen $2^4 = 16$ funciones binarias posibles, pero sólo algunas tendrán algún interés. Estas son las operaciones binarias que destacaremos:

x	y	OR	AND	XOR	NOR	NAND	EQ
0	0	0	0	0	1	1	1
0	1	1	0	1	0	1	0
1	0	1	0	1	0	1	0
1	1	1	1	0	0	0	1

Simplificación

Notemos que una función $f(x, y) = NOT\ x\ AND\ NOT\ y$, tiene la siguiente tabla de verdad y es equivalente a la función que hemos denotado como NOR .

x	y	NOT x	NOT y	NOT x AND NOT y
0	0	1	1	1
0	1	1	0	0
1	0	0	1	0
1	1	0	0	0

¿Por qué es importante la simplificación?