

APÉNDICE A: TABLAS DE VERDAD EN MS EXCEL

MS Excel es una herramienta adecuada para manejar tablas de verdad. Pueden definirse tablas relativamente grandes, copiar y pegar patrones que se repiten, mirar partes de una tabla grande, etc.

Sin embargo, hay algunos detalles que hacen que las cosas puedan no ser tan directas. A continuación se presenta una alternativa para diseñar tablas de verdad y se dan algunos consejos para que sean más sencillas de construir.

A.1 LA LÓGICA DE EXCEL

En Excel los nombres de las funciones dependen de la forma en que se instala el paquete. Para empezar, las funciones tienen un nombre en las versiones inglesas diferente del usado en las versiones españolas (y claro, en otros idiomas). Esta decisión de diseño de los inventores de MS Excel hace que las fórmulas deban traducirse entre diferentes idiomas, Más aun: los operadores lógicos no tiene nombres estándar. En lo que sigue se va a suponer que se tiene una instalación de Excel en español.

Excel cuenta con las siguientes operaciones lógicas:

Notación Excel	Notación lógica	Interpretación
FALSO	false	valor de verdad "falso"
VERDADERO	true	valor de verdad "verdadero"
NO(x)	$\neg x$	negación de x
Y(x,y)	$x \wedge y$	conjunción de x e y
O(x,y)	$x \vee y$	disyunción de x e y

Los operandos x e y son referencias a celdas de una hoja Excel¹, v.gr., A23.

Estas funciones dan para construir todas las tablas de verdad que uno conciba. La razón es que se sabe que el conjunto de operadores básicos de Excel es *completo*, en el sentido que los demás operadores binarios se pueden simular con una fórmula Excel.

Por ejemplo:

Notación Excel	Notación lógica	Operación simulada
$O(NO(x), y)$	$\neg x \vee y$	$x \Rightarrow y$
$Y(O(NO(x), y), O(NO(y), x))$	$(\neg x \vee y) \wedge (\neg y \vee x)$	$x \equiv y$
$NO(Y(O(NO(x), y), O(NO(y), x)))$	$\neg((\neg x \vee y) \wedge (\neg y \vee x))$	$x \neq y$

Estos son solo ejemplos de cómo se hace: puede haber formas más simples de hacer estas simulaciones.

¹ En realidad, en Excel se pueden definir nombres para celdas, que reflejen la semántica del dato que éstas contienen, v.gr., una celda puede llamarse x , y otra y . No se va a suponer que se aprovecha esta posibilidad, porque no siempre funciona como se espera en lo que adelante se va a explicar.

A.2 PARA HACER UNA TABLA DE VERDAD

Se puede usar lo básico, como ya se indicó. A continuación se muestra una tabla de verdad que ilustra por qué el operador de equivalencia es asociativo:

p	q	r	(p EQ q) EQ r	p EQ (q EQ r)	((p EQ q) EQ r) EQ (p EQ (q EQ r))
FALSO	FALSO	FALSO	FALSO	FALSO	VERDADERO
FALSO	FALSO	VERDADERO	VERDADERO	VERDADERO	VERDADERO
FALSO	VERDADERO	FALSO	VERDADERO	VERDADERO	VERDADERO
FALSO	VERDADERO	VERDADERO	FALSO	FALSO	VERDADERO
VERDADERO	FALSO	FALSO	VERDADERO	VERDADERO	VERDADERO
VERDADERO	FALSO	VERDADERO	FALSO	FALSO	VERDADERO
VERDADERO	VERDADERO	FALSO	FALSO	FALSO	VERDADERO
VERDADERO	VERDADERO	VERDADERO	VERDADERO	VERDADERO	VERDADERO

Se usa la notación .EQ. para el operador \equiv . Las fórmulas que se escriben en cada casilla son construidas de acuerdo con lo explicado en 1.

A.3 CÓMO HACERLO MÁS CÓMODO

En las tablas definidas como en A.2 se tiene una incomodidad evidente. No parecen prácticas, por ser poco legibles, en algún sentido.

Para empezar, se prefiere escribir 1 en lugar de VERDADERO y 0 en lugar de FALSO. Pero, al pretender hacer este cambio, las funciones lógicas básicas dejan de calcular bien. Sin embargo, las operaciones lógicas se puede similar con aritmética, de acuerdo con la siguiente tabla (si α es una expresión booleana, $e(\alpha)$ es su representación en Excel):

Expresión booleana	Excel
false	0
true	1
$\neg\alpha$	1 - e(α)
$\alpha \wedge \beta$	e(α)*e(β)
$\alpha \vee \beta$	max(e(α), e(β))
$\alpha \Rightarrow \beta$	si(e(α)=1, e(β), 1-e(β))
$\alpha \equiv \beta$	e(α)=e(β)
$\alpha \neq \beta$	e(α)<>e(β)

La traducción no es, necesariamente, de la forma indicada. Por ejemplo, también se puede traducir $\alpha \vee \beta$, como $1 - (1 - e(\alpha)) * (1 - e(\beta))$.

Una posibilidad adicional es usar funciones de Visual Basic que simulen las operaciones. Por ejemplo, la función EQ simula el operador de equivalencia:

```
Function EQ(x, y)
  If x = y Then
    EQ = 1
  Else
    EQ = 0
  End If
End Function
```

Con mejoras como las anteriores se pueden construir tablas de verdad legibles y fáciles de manipular. Por ejemplo, la asociatividad de la equivalencia:

p	q	r	(p EQ q) EQ r	p EQ (q EQ r)	((p EQ q) EQ r) EQ (p EQ (q EQ r))
0	0	0	0	0	1
0	0	1	1	1	1
0	1	0	1	1	1
0	1	1	0	0	1
1	0	0	1	1	1
1	0	1	0	0	1
1	1	0	0	0	1
1	1	1	1	1	1