



# PRÁCTICA Nº4

## Lógica con el Mathematica (PARTE I) *Conectivas y tablas de verdad.*

### 1.- INTRODUCCIÓN.

Con esta práctica pretendemos realizar los ejercicios vistos en teoría utilizando el programa Mathematica. Veremos como se usan las conectivas mediante este programa, y por último realizaremos un pequeño programa que calcule tablas de verdad.

### 2.- CONECTIVAS Y FORMAS ENUNCIATIVAS.

Recordemos que las conectivas más usuales que conocemos son:

- 1.-  $\sim$  negación
- 2.-  $\wedge$  y
- 3.-  $\vee$  o
- 4.-  $\rightarrow$  entonces
- 5.-  $\leftrightarrow$  si y sólo si

además existen otras como el NOR, NAND y el XOR. Veamos como se consiguen todas ellas usando el Mathematica:

CONECTIVA	FORMA 1	FORMA 2	FORMA 3
$\sim p$	$!p$	Not[p]	$\backslash[\text{Not}]p$
$p \wedge q$	$p \& \& q$	And[p,q]	$p \backslash[\text{And}] q$
$p \vee q$	$p    q$	Or[p,q]	$p \backslash[\text{Or}] q$
$p \rightarrow q$	-	Implies[p,q]	$p \backslash[\text{Implies}] q$

$p \leftrightarrow q$  no lo podemos conseguir directamente, por tanto vamos a definir una función que llamaremos  $Sii[p,q]$  que haga lo mismo, para ello tengamos en cuenta que las formas enunciativas:

$$p \leftrightarrow q$$

y

$$(p \rightarrow q) \wedge (q \rightarrow p)$$

son equivalentes. Por tanto la definiremos como sigue:

**Sii[p\_,q\_]:=Implies[p,q] && Implies[q,p]**

Podría definirse de otra forma:

**Sii2[p\_,q\_]:=Which[p==q, True, p!=q, False]**

Para escribir formas enunciativas, tan sólo tendremos que traducir al Mathematica las expresiones correspondientes cambiando las conectivas como se ha indicado antes, por ejemplo la forma enunciativa  $A = (p \wedge q \vee (\sim r)) \rightarrow (r \wedge q)$  quedaría:

**A:=Implies[p && (q || (!r)), r && q]**

Si queremos evaluar la forma enunciativa para una combinación de valores utilizaremos la función **TrueQ[-]**, y escribiremos **TrueQ[A]** o directamente pediremos el valor de A. Por ejemplo:

**p=True;**  
**q=False;**  
**r=True;**  
**TrueQ[A]**    (*o directamente escribiremos A*)

Ejercicio 1:

- a) Escribir funciones que representen las conectivas: NOR, NAND y XOR.
- b) Para p = Verdad, q = Verdad, r = Falso y s = Verdad, evaluar la forma enunciativa:  $B = (p \leftrightarrow q) \downarrow (r \uparrow s)$ .

**Solución:**

### 3.- TABLAS DE VERDAD.

Una vez que sabemos la forma de representar las conectivas, o de definirlas, el siguiente paso que tendremos que dar será escribir un programa que calcule las tablas de verdad de las formas enunciativas. Para aprender como se pueden calcular las tablas de verdad con Mathematica, vamos a resolver un ejemplo.

*Ejemplo 1:*

Calcular la tabla de verdad de la forma enunciativa:  $(p_1 \leftrightarrow (p_2 \wedge p_3))$ .

SOLUCIÓN:

PROGRAMA	COMENTARIOS
<b>Sii[a_,b_] := Implies[a,b] &amp;&amp; Implies[b,a];</b>	Zona para incluir las definiciones de conectivas que necesitemos.
<b>n=3;</b>	Nº de variables de enunciado.
<b>p=Table[False,{t,n}];</b>	Declaración de las variables de enunciado.
<b>expresion:=Sii[p[[1]],p[[2]] &amp;&amp; p[[3]]];</b>	Forma enunciativa cuya tabla de verdad queremos calcular.
<b>tabla=Table["F",{x,(2^n+1)},{y,n+1}];</b>	Declaración de la tabla donde introduciremos los valores resultantes.
<b>For[k=1,k&lt;n+1,k++,tabla[[1,k]] = StringJoin["p",ToString[k]]];</b>	Bucle que introduce los valores en la primera fila de la tabla que indicarán a que variable de enunciado nos referimos en cada columna.
<b>tabla[[1,n+1]] = "Exp";</b>	Da el valor "exp" a la columna que dará los valores que resultan en la forma enunciativa.
<b>For[i=0,i&lt;2^n,i++, j=i; For[f=n,f&gt;0,f--, resto=Mod[j,2]; j=Floor[j/2]; If[resto==0,p[[f]]=True; tabla[[i+2,f]] = "V",p[[f]]=False]; ]; If[TrueQ[expresion],tabla[[i+2,n+1]] = "V"]; ];</b>	<div>Bucle que calcula la combinación en cada caso.</div> <div>Bucle que recorre todas las posibles combinaciones en la tabla de verdad.</div>
<b>TableForm[tabla]</b>	Muestra en pantalla la tabla de verdad.

En primer lugar, debemos incluir las definiciones de aquellas conectivas que no están integradas en Mathematica y que pensamos usar, en el ejemplo incluimos la definición de la conectiva  $\leftrightarrow$ .

La zona sombreada con gris claro nos resalta las líneas que son especiales para el cálculo de esta forma enunciativa en particular. La primera línea sombreada:

**n=3;**

define el número de variables de enunciado que contiene nuestra forma enunciativa, en este caso 3. La segunda línea sombreada:

**expresion=Sii[p[[1]],p[[2]] && p[[3]]];**

introduce en el ordenador la forma enunciativa con el nombre ‘expresión’, cuya tabla de verdad queremos calcular. Nótese que cada conectiva ha sido sustituida por su definición o símbolo correspondiente, tal y como se vio en el apartado 3. Por otra parte las variables de enunciado se escriben: p[[1]] para  $p_1$ , p[[2]] para  $p_2$  y p[[3]] para  $p_3$ ; esto debe de ser así porque todas ellas se declaran mediante un vector n-dimensional, donde n aquí es 3, para más detalle véase la práctica correspondiente.

En el programa también tenemos destacados los dos bucles anidados que realizan los cálculos de la tabla de verdad, como podemos observar, uno de ellos recorre las  $2^n$  posibles combinaciones de valores, en este caso 8, mientras que el otro calcula la combinación correspondiente a cada caso, con este fin, recorre las n variables de enunciado de nuestra forma enunciativa, en este caso 3.

Tras copiar el programa y presionar MAY.+INTRO, la salida que nos dará el Mathematica será:

p1	p2	p3	Exp
V	V	V	V
V	V	F	F
V	F	V	F
V	F	F	F
F	V	V	F
F	V	F	V
F	F	V	V
F	F	F	V

### Ejercicio 2:

Calcula las tablas de verdad de las siguientes formas enunciativas:

- a)  $(\sim r \vee p) \rightarrow q$ .
- b)  $(p \rightarrow q) \wedge r$ .
- c)  $[(p \rightarrow q) \wedge r] \rightarrow [(\sim r \vee p) \rightarrow q]$ .

### Solución:

