

# Notas de Matematica Discreta

Ivan Litteri

## 1 Lógica

Las reglas de la logica le dan un significado preciso a los enunciados matematicos o sentencias matematicas. Estas reglas se usan para distinguir entre argumentos validos y no validos. La logica tiene ademas numerosas aplicaciones en ciencias de la computacion. Las reglas de la logiuca se usan en el diseño de circuitos de ordenador, la construccion de programas informaticos, la verificacion de que un programa esta bien construido y mas.

### Proposicion

Una proposicion es una oracion declarativa que es correcta o falsa, pero no ambas cosas a la vez.

### Ejemplos

1. Todas las siguientes oraciones declarativas son proposiciones:

- (a) Bruselas es la capital de la Union Europea.
- (b) Toronto es la capital de Canada.
- (c)  $1 + 1 = 2$ .
- (d)  $2 + 2 = 3$ .

Las proposiciones 1 y 3 son correctas, mientras que la 2 y 4 son falsas.

2. Considera las siguientes oraciones:

- (a) ¿Que hora es?
- (b) Lee esto con atencion.
- (c)  $x + 1 = 2$ .
- (d)  $x + y = z$ .

Las frases 1 y 2 no son proposiciones porque no son declarativas. Las frases 3 y 4 no son proposiciones porque no son ni verdaderas ni falsas, ya que no se les han asignado valores a las variables.

El **valor de verdad** de una proposicion es  $V$  si es verdadera y  $F$  si es falsa.

Se llama **calculo proposicional** o **logica proposicional** al area de la logica que trata de proposiciones. Las nuevas proposiciones, llamadas **proposicion compuestas** o **proposiciones compuestas**, se forman a partir de las existentes usando operadores logicos.

### Tabla de Verdad

Una **tabla de verdad** muestra las relaciones entre los valores de verdad ed proposiciones. Son especialmente valiosas a la hora de determinar los valores de verdad de proposiciones construidas a partir de proposiciones simples.

## Proposiciones Compuestas

**Definición.** Sea  $p$  una proposición. El enunciado

*No se cumple  $p$ ,*

es otra proposición, llamada la *negación* de  $p$ . La negación de  $p$  se denota mediante  $p'$  o  $(/p$  o  $\neg p)$ . La proposición  $p'$  se lee *no  $p$* .

La negación de una proposición se puede considerar como el resultado de aplicar el **operador negación** sobre una proposición. El operador negación construye una nueva proposición a partir de la proposición individual existente.

$p$	$p'$
0	1
1	0

**Definición.** Sean  $p$  y  $q$  proposiciones. La proposición  *$p$  y  $q$* , denotada por  $p \wedge q$ , es la proposición que es verdadera cuando tanto  $p$  como  $q$  son verdaderas y falsa en cualquier otro caso. La proposición  $p \wedge q$  se llama *conjunción* de  $p$  y  $q$ .

$p$	$q$	$p \wedge q$
0	0	0
0	1	0
1	0	0
1	1	1

**Definición.** Sean  $p$  y  $q$  proposiciones. La proposición  *$p$  o  $q$* , denotada por  $p \vee q$ , es la proposición que es falsa cuando tanto  $p$  como  $q$  son falsas y verdadera en cualquier otro caso. La proposición  $p \vee q$  se llama *disyunción* de  $p$  o  $q$ .

$p$	$q$	$p \vee q$
0	0	0
0	1	1
1	0	1
1	1	1

**Definición.** Sean  $p$  y  $q$  proposiciones. El conectivo lógico *o exclusivo* de  $p$  y  $q$ , denotada por  $p \oplus q$ , es la proposición que es verdadera cuando exactamente una de las proposiciones  $p$  y  $q$  es verdadera y falsa en cualquier otro caso.

$p$	$q$	$p \oplus q$
0	0	0
0	1	1
1	0	1
1	1	0

## Implicaciones

El concepto matematico de implicacion es independiente de la relacion causa-efecto entre hipotesis y conclusion. Especifica valores de verdad, no se basa en el uso del lenguaje.

**Definicion.** Sean  $p$  y  $q$  proposiciones. La *implicacion*  $p \rightarrow q$  es la proposicion que es falsa cuando  $p$  es verdadera y  $q$  es falsa, y verdadera en cualquier otro caso. En esta implicacion  $p$  se llama *hipotesis* o *antecedente* o *premisa* y  $q$  se llama *tesis* o *conclusion* o *consecuencia*.

$p$	$q$	$p \rightarrow q$
0	0	1
0	1	1
1	0	0
1	1	1

Las declaraciones condicionales (como las implicaciones) tienen un rol esencial en el razonamiento matematico, existe una variedad de terminologias para expresar  $p \rightarrow q$ . Entre ellas:

"si $p$ , entonces $q$ "	" $p$ implica $q$ "
"si $p$ , $q$ "	" $p$ solo si $q$ "
" $p$ es suficiente para $q$ "	"una condicion suficiente para $q$ es $p$ "
" $q$ si $p$ "	" $q$ cuando sea $p$ "
" $q$ cuando $p$ "	" $q$ es necesario para $p$ "
"una condicion necesaria para $p$ es $q$ "	" $q$ sigue de $p$ "
" $q$ a menos que $p$ "	" $q$ siempre que $p$ "

## Reciproca, contrarreciproca e inversa

Hay algunas implicaciones relacionadas con  $p \rightarrow q$  que pueden formarse a partir de ella. La proposicion  $q \rightarrow p$  se llama **reciproca** de  $p \rightarrow q$ . La **contrarreciproca** de  $p \rightarrow q$  es  $q' \rightarrow p'$ . La proposicion  $p' \rightarrow q'$  es la **inversa** de  $p \rightarrow q$ . Cuando dos proposiciones compuestas tienen siempre los mismos valores de verdad las llamamos **equivalentes**, de tal forma que una implicacion y su contrarreciproca son equivalentes. La reciproca y la inversa de una implicacion tambien son equivalentes.

**Definicion.** Sean  $p$  y  $q$  proposiciones. La *bicondicional*, o *doble impliacion*,  $p \leftrightarrow q$  es la proposicion que es verdadera cuando  $p$  y  $q$  tienen los mismos valores de verdad y falsa en los otros casos.

$p$	$q$	$p \leftrightarrow q$
0	0	1
0	1	0
1	0	0
1	1	1

Notese que  $p \leftrightarrow q$  es verdadera cuando  $p \rightarrow q$  y  $q \rightarrow p$  son verdaderas y falso de otra manera. Las formas mas comunes de expresar esto es:

" $p$  es necesario y suficiente para  $q$ "  
 "si  $p$  entonces  $q$ , y biceversa"  
 " $p$  sii  $q$ ". " $p$  exactamente cuando  $q$ "

## Precedencia de operadores logicos

Operador	Precedencia
' (not)	1
$\wedge$	2
$\vee$	4
$\rightarrow$	4
$\leftrightarrow$	5

## Logica y Operaciones con Bits

Un **bit** tiene dos valores posibles: 0 y 1. La palabra bit viene de la expresion inglesa *binary digit*. Un bit se puede utilizar para represenar un valor de verdad. Se usa 1 para representar V de verdadero y 0 para representar F de falso. Una variable se llama **variable booleana** si su valor es verdadero o falso. Se puede representar una variable booleana con bits.

**Definicion.** Una *cadena de bits* es una sucesion de cero o mas bits. La longitud de esta cadena es el numero de bits de la cadena.

## Traducir Oraciones

Traducir oraciones del lenguaje natural a expresiones legicas es una parte esencial de especificar sistemas tantohardware como software.

Las especificaciones de sistema deben ser **consistentes**, esto es, no deben contener recursos conflictivos requisitos que podrían utilizarse para derivar una contradicción. Cuando las especificaciones no son consistentes, no habría forma de desarrollar un sistema que satisfaga todas las especificaciones.

## Busquedas Booleanas

En las busquedas booleanas se usa la conexion *AND* para emparejar datos almacenados que contengan los dos terminos de la busqueda, la conexion *OR* se usa para emparejar uno o ambos terminos de la busqueda y la conexion *NOT* (a veces escrita *ANDNOT*) se usa para excluir un termino particular de busqueda.

## Circuitos Logicos

Un **circuito logico** (o **circuito digital**) recibe señales de entrada  $p_1, \dots, p_n$ , cada bit [0 (off) o 1 (on)], y produce una señal de salida  $s_1, \dots, s_n$  para cada bit.

Circuitos digitales complejos pueden contruirse a partir de tres circuitos basicos llamados **compuertas**. El **inversor**, o **compuerta NOT**, toma el bit de entrada  $p$ , y produce una salida  $p'$ . La **compuerta OR** toma dos entradas  $p$  y  $q$ , cada una un bit, y produce una señal de salida  $p \vee q$ . Finalmente, la **compuerta AND** toma dos entradas  $p$  y  $q$ , cada una un bit, y produce una señal de salida  $p \wedge q$ .

### 1.1 Equivalencias proposicionales

**Definicion.** Una proposicion compuesta que es siempre verdadera, no importa los valores de verdad de las proposiciones que la componen, se denomina *tautologia*. Una proposicion compuesta que es siepre falsa se

denomina *contradiccion*. Finalmente, una proposicion que no es ni una tautologia ni una contradiccion se denomina *contingencia*.

## Equivalencias Logicas

Las proposiciones compuestas que tienen los mismos valores de verdad en todos los casos posibles se llaman **logicamente equivalentes**.

**Definicion.** Se dice que las proposiciones  $p$  y  $q$  son *logicamente equivalentes* si  $p \leftrightarrow q$  que es una tautologia. La notacion  $p \equiv q$  denota que  $p$  y  $q$  son logicamente equivalentes.

## Equivalencias Logicas

<i>Equivalencia</i>	<i>Nombre</i>
$p \wedge V \equiv p$ $p \vee F \equiv p$	Leyes de identidad
$p \vee V \equiv V$ $p \wedge F \equiv F$	Leyes de dominacion
$p \vee p \equiv p$ $p \wedge p \equiv p$	Leyes de idempotentes
$(p')' \equiv p$	Ley de la doble negacion
$p \vee q \equiv q \vee p$ $p \wedge q \equiv q \wedge p$	Leyes de conmutativas
$(p \vee q) \vee r \equiv p \vee (q \vee r)$ $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$	Leyes de asociativas
$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$ $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	Leyes de distributivas
$(p \wedge q)' \equiv p' \vee q'$ $(p \vee q)' \equiv p' \wedge q'$	Leyes de Morgan
$p \vee (p \wedge q) \equiv p$ $p \wedge (p \vee q) \equiv p$	Leyes de absorcion
$p \vee p' \equiv V$ $p \wedge p' \equiv F$	Leyes de negacion

<i>Equivalencia</i>
$p \rightarrow q \equiv p' \vee q$ $p \rightarrow q \equiv q' \vee p$ $p \vee q \equiv p' \rightarrow q$ $p \wedge q \equiv (p \vee q')'$ $(p \vee q') \equiv p \rightarrow q'$ $(p \rightarrow q) \wedge (p \rightarrow r) \equiv p \rightarrow (q \wedge r)$ $(p \rightarrow r) \wedge (q \rightarrow r) \equiv (p \vee q) \rightarrow r$ $(p \rightarrow q) \vee (p \rightarrow r) \equiv p \rightarrow (q \vee r)$ $(p \rightarrow r) \vee (q \rightarrow r) \equiv (p \wedge q) \rightarrow r$
$p \leftrightarrow q \equiv (p \rightarrow q) \wedge (q \rightarrow p)$ $p \leftrightarrow q \equiv p' \leftrightarrow q'$ $p \leftrightarrow q \equiv (p \wedge q) \vee (p' \wedge q')$ $(p \leftrightarrow q)' \equiv p \leftrightarrow q'$

## Satisfaccion

Una proposición compuesta es **satisfactoria** si hay una asignación de valores de verdad a sus variables que la hace verdadera (es decir, cuando es una tautología o una contingencia). Cuando no existen tales asignaciones, es decir, cuando la proposición compuesta es falsa para todas las asignaciones de valores de verdad a sus variables, la proposición compuesta es **insatisfactorio**. Nótese que una proposición compuesta no es satisfactoria si y solo si su negación es verdadera para todas las asignaciones de valores de verdad a las variables, es decir, si y solo si su negación es una tautología.

Cuando encontramos una asignación particular de valores de verdad que hace que una proposición compuesta sea verdadera, hemos demostrado que es satisfactoria; tal asignación se denomina solución de este problema de satisfacibilidad particular. Sin embargo, para mostrar que una proposición compuesta es insatisfactoria, necesitamos mostrar que toda asignación de valores de verdad a sus variables la hace falsa. Aunque siempre podemos usar una tabla de verdad para determinar si una proposición compuesta es satisfactoria, a menudo es más eficiente no hacerlo.

## 1.2 Predicados y cuantificadores

### Cuantificadores

Cuando todas las variables de una funcion proposicional se le han asignado valores, la sentencia resultante se convierte en una proposicion con un cierto valor de verdad. No obstante, hay otra forma importante, llamada **cuantificacion**, de crear una proposicion a partir de una funcion proposicional. Tenemos la cuantificacion universal y la cuantificacion existencial.

#### El cuantificador universal

Muchas sentencias matematicas imponen que una propiedad es verdadera para todos los valores de una variable en un dominio particular, llamado el **universo de discurso** o **dominio**. Tales sentencias se expresan utilizando un cuantificador universal. La cuantificacion universal de una funcion proposicional es la proposicion que afirma que  $P(x)$  es verdadera para todos los valores de  $x$  en el dominio. EL dominio especifica los posibles valores de la variable  $x$ .

#### Definicion 1

La *cuantificacion universal* de  $P(x)$  es la proposicion  $P(x)$  es verdadera para todos los valores  $x$  del dominio.

La notacion  $\forall xP(x)$  denota la cuantificacion universal de  $P(x)$ . Aqui llamaremos al simbolo  $\forall$  el **cuantificador universal**. La proposicion  $\forall xP(x)$  se lee como *para todo  $xP(x)$*  o *para cada  $xP(x)$*  o *para cualquier  $xP(x)$* .

Para mostrar que una sentencia de la forma  $\forall xP(x)$  es falsa, donde  $P(x)$  es una funcion proposicional, solo necesitamos encontrar un valor de  $x$  del dominio para el cual  $P(x)$  sea falsa. Este valor de  $x$  se llama **contraejemplo** de la sentencia  $\forall xP(x)$ .

#### El cuantificador existencial

Muchas sentencias matematicas afirman que hay un elemento con una cierta propiedad. Tales sentencias se expresan mediante cuantificadores existenciales. Con un cuantificador existencial formamos una proposicion que es verdadera si y solo si  $P(x)$  es verdadera para al menos un valor de  $x$  en el dominio.

#### Definicion 2

La *cuantificacion existencial* de  $P(x)$  es la proposicion *Existe un elemento  $x$  en el dominio tal que  $P(x)$  es verdadera*.

Usamos la notacion  $\exists xP(x)$  para la cuantificacion existencial de  $P(x)$ . El simbolo  $\exists$  se denomina **cuantificador existencial**. La cuantificacion existencial  $\exists xP(x)$  se lee como *Hay un  $x$  tal que  $P(x)$*  o *Hay al menos un  $x$  tal que  $P(x)$*  o *Para algun  $xP(x)$*

### Cuantificadores

Sentencia	¿Cuando es verdadera?	¿Cuando es falsa?
$\forall xP(x)$	$P(x)$ es verdadera para todo $x$	Hay un $x$ para el que $P(x)$ es falsa
$\exists xP(x)$	Hay un $x$ para el que $P(x)$ es verdadera	$P(x)$ es falsa para todo $x$

Cuando se quiere determinar el valor de verdad de una cuantificacion, a veces es util realizar una busqueda sobre todos los posibles valores del dominio. Supongamos que hay  $n$  objetos en el dominio de la variable  $x$ . Para determinar si  $\forall xP(x)$  es verdadera para todos ellos. Si encontramos un valor de  $x$  para el cual  $P(x)$  es falsa, habremos demostrado que  $\forall xP(x)$  es falsa. En caso contrario,  $\forall xP(x)$  es verdadera. Para ver si

$\exists xP(x)$  es verdadera, barremos los  $n$  posibles de  $x$  buscando algun valor para el cual  $P(x)$  sea verdadera. Si encontramos uno, entonces  $\exists xP(x)$  es verdadera. Si no encontramos tal valor de  $x$ , habremos determinado que  $\exists xP(x)$  es falsa.

### Variables ligadas

Cuando un cuantificador se usa sobre la variable  $x$  o cuando asignamos un valor a esta variable, decimos que la variable aparece **ligada**. Una variable que no aparece ligada por un cuantificador o fijada a un valor particular, se dice que es **libre**. Todas las variables que aparecen en una funcion proposicional deben ser ligadas para convertirla en proposicion. Esto se puede hacer utilizando una combinacion de cuantificadores universales, cuantificadores existenciales y asignacion de valores.

La parte de una expresion logica a la cual se aplica el cuantificador se llama **ambito** de este cuantificador. Consecuentemente, una variable es libre si esta fuera del ambito de todos los cuantificadores en la proposicion compuesta.

### Negaciones

Cuando el dominio de un predicado  $P(x)$  consiste en  $n$  elementos, donde  $n$  es un entero positivo, las reglas de la negcion de sentencias cuantificadas son exactamente las mismas que las leyes de De Morgan. Esto es asi porque  $(\forall xP(x))'$  es lo mismo que  $(P(x_1) \wedge P(x_2) \wedge \dots \wedge P(x_n))'$ , equivalente a  $P(x_1)' \vee P(x_2)' \vee \dots \vee P(x_n)'$  por las leyes de De Morgan. Esto es lo mismo que  $\exists xP(x)'$ . De forma analoga,  $(\exists xP(x))'$  es lo mismo que  $(P(x_1) \vee P(x_2) \vee \dots \vee P(x_n))'$ , equivalente a  $P(x_1)' \wedge P(x_2)' \wedge \dots \wedge P(x_n)'$  por las leyes de De Morgan, lo que equivale  $\forall xP(x)'$ .

### Cuantificadores

<i>Negacion</i>	<i>proposicion compuesta equivalente</i>	<i>¿Cuando es verdadera la negacion?</i>	<i>¿Cuando es falsa?</i>
$(\forall xP(x))'$	$\forall xP(x)'$	Para cada $x$ , $P(x)$ es falsa	Hay un $x$ para el que $P(x)$ es verd
$(\exists xP(x))'$	$\exists xP(x)'$	Hay un $x$ para el que $P(x)$ es falsa	$P(x)$ es verdadera para cada $x$

## 1.3 Cuantificadores anidados

Son cuantificadores que se localizan dentro del rango de aplicacion de otros cuantificadores, como en la sentencia  $\forall x\exists y(x + y = 0)$ . Los cuantificadores anidados se usan tanto en matematicas como en ciencias de la computacion.

### Formalizacion de sentencias con cuantificadores anidados

#### Ejemplos

- Supongamos que el dominio de las variables reales  $x$  e  $y$  consiste en todos los reales. La sentencia

$$\forall x\forall y(x + y = y + x)$$

afirma que  $x + y = y + x$  para todo par de numeros reales  $x$  e  $y$ . Es la ley conmutativa para la suma de los numeros reales. De la misma forma la sentencia

$$\forall x\exists y(x + y = 0)$$

afirma que para cada numero real  $x$  hay un real  $y$  tal que  $x + y = 0$ . Esto declara que todo numero real tiene un inverso para la suma. Analogamente, la sentencia

$$\forall x\forall y\forall z(x + (y + z) = (x + y) + z)$$

es la ley asociativa para la suma de numeros reales.

2.  $\forall x \forall y ((x > 0) \wedge (y < 0) \rightarrow (xy < 0)) \equiv$  todo numero  $x$  positivo y para todo  $y$  negativo, el producto de ese numero  $x$  por  $y$  sera menor a cero.
3.  $\forall x (C(x) \vee \exists y (C(y) \wedge F(x, y))) \therefore C(x) = x$  tiene una computadora,  $F(x, y) = x$  e  $y$  son amigos y el dominio tanto para  $x$  como para  $y$  consiste en todos los estudiantes de tu facultad. Nos dice que para cada estudiante  $x$  de tu facultad,  $x$  tiene una computadora o hay un estudiante  $y$  tal que  $y$  tiene una computadora y  $x$  e  $y$  son amigos. Con otras palabras, todo estudiante de tu facultad tiene una computadora o un amigo que tiene una.

## Formalizacion de sentencias en expresiones logicas

### Ejemplos

1. Expresa la sentencia «Si una persona es del sexo femenino y tiene un hijo, esta persona es la madre de alguien» como una expresión lógica que involucre predicados, cuantificadores –cuyo dominio es el conjunto de todas las personas– y conectivos lógicos.

*Solucion:* La frase anterior se puede expresar como «Para toda persona  $x$ , si la persona  $x$  es del sexo femenino y la persona  $x$  tiene un hijo, entonces existe una persona  $y$  tal que  $x$  es madre de la persona  $y$ ». Introducimos los predicados  $F(x)$  para representar « $x$  es del sexo femenino»,  $P(x)$  para representar « $x$  tiene un hijo» y  $M(x, y)$  para representar « $x$  es madre de  $y$ ». La frase original se puede expresar como

$$\forall x ((F(x) \wedge P(x)) \rightarrow \exists y M(x, y))$$

$$\forall x \exists y ((F(x) \wedge P(x)) \rightarrow M(x, y))$$

2. «La suma de dos enteros positivos es positiva»  $\equiv \forall x \forall y ((x > 0) \wedge (y > 0) \rightarrow (x + y > 0))$
3. «Todo numero real, excepto cero, tiene un inverso para el producto»  $\equiv \forall x \exists y ((x \neq 0) \rightarrow (xy = 1))$
4. Definicion de limita usando cuantificadores

$$\lim_{x \rightarrow a} f(x) = L$$

Tenemos que para todo numero real  $\varepsilon > 0$ , existe un numero real  $\delta > 0$  tal que  $|f(x) - L| < \varepsilon$  siempre que  $0 < |x - a| < \delta$ .

$$\forall \varepsilon \exists \delta (0 < |x - a| < \delta \rightarrow |f(x) - L| < \varepsilon)$$

o

$$\forall \varepsilon > 0 \exists \delta > 0 \forall x (0 < |x - a| < \delta \rightarrow |f(x) - L| < \varepsilon)$$

## Negacion de cuantificadores anidados

Las sentencias con varios cuantificadores anidados se pueden negar aplicando sucesivamente las reglas de negación de las sentencias que contienen un único cuantificador.

### Ejemplos

1. Negar  $\forall x \exists y (xy = 1)$  de tal forma que ninguna negacion preceda al cuantificador.

*Solucion:*

$$(\forall x \exists y (xy = 1))' \equiv \forall x (\exists y (xy = 1))' \equiv \forall x \exists y (xy = 1)'$$

$$\therefore (xy = 1)' \equiv (xy \neq 1)$$

$$\Rightarrow \forall x \exists y (xy \neq 1)$$



2. Usa cuantificadores para expresar la sentencia «No existe ninguna mujer que haya viajado en un vuelo de cada una de las líneas aéreas del mundo».

*Solucion:*

$$\begin{aligned}
 (\forall w \forall a \exists f (P(w, f) \wedge Q(f, a)))' &\equiv \forall w' \forall a \exists f (P(w, f) \wedge Q(f, a)) \\
 &\equiv \forall w \forall a' \exists f (P(w, f) \wedge Q(f, a)) \\
 &\equiv \forall w \forall a \exists f' (P(w, f) \wedge Q(f, a)) \\
 &\equiv \forall w \forall a \exists f (P(w, f) \wedge Q(f, a))' \\
 &\equiv \forall w \forall a \exists f (P(w, f)' \wedge Q(f, a)')
 \end{aligned} \tag{1}$$

«Para toda mujer hay una línea aérea tal que, para todo vuelo, esta mujer no ha viajado en ese vuelo o ese vuelo no es de esa línea aérea».

### 1.3.1 El orden de los cuantificadores

Cuantificadores de dos variables

<i>Sentencia</i>	<i>¿Cuando es verdadera?</i>	<i>¿Cuando es falsa?</i>
$\forall x \forall y P(x, y)$	$P(x, y)$ es verdadera para todo $x, y$	Hay un par $x, y$ para el cual $P(x, y)$ es falsa
$\forall x \exists y P(x, y)$	Para todo $x$ hay un $y$ para el cual $P(x, y)$ es verdadera	Hay un $x$ tal que $P(x, y)$ es falsa para todo $y$
$\exists x \forall y P(x, y)$	Hay un $x$ tal que $P(x, y)$ es verdadera para todo $y$	Para todo $x$ hay un $y$ para el cual $P(x, y)$ es falsa
$\exists x \exists y P(x, y)$	Hay un par $x, y$ para el cual $P(x, y)$ es verdadera	$P(x, y)$ es falsa para todo $x, y$

De forma similar para determinar si  $\forall x \exists y P(x, y)$  es verdadera, recorreremos en un bucle todos los valores de  $x$ . Para cada  $x$ , recorreremos en un bucle los valores de  $y$  hasta que encontramos un  $y$  para el cual  $P(x, y)$  es verdadera. Si para todos los  $x$  encontramos tal valor de  $y$ , entonces  $\forall x \exists y P(x, y)$  es verdadera; si para algun  $x$  no encontramos un valor de  $y$  con esa propiedad, entonces  $\forall x \exists y P(x, y)$  falsa. Para ver si  $\exists x \forall y P(x, y)$  es verdadera, recorreremos los valores de  $x$  en un bucle hasta que encontramos un  $x$  para el cual  $P(x)$  es siempre verdadera cuando recorremos en un bucle todos los valores de  $y$ . Una vez encontrado tal valor de  $x$ , sabemos que  $\exists x \forall y P(x, y)$  es verdadera. Si no encontramos nunca un  $x$  como ese, entonces sabremos que  $\exists x \forall y P(x, y)$  es falsa. Finalmente, para saber si  $\exists x \exists y P(x, y)$  es verdadera, recorreremos en un bucle los valores de  $x$ , y para cada valor de  $x$  recorreremos los valores de  $y$  hasta que encontremos un  $x$  oara el cual haya un  $y$  que verifique que  $P(x, y)$  sea verdadera.