

Capítulo 1

Lógica Matemática

Resultado de aprendizaje

Diseña soluciones a problemas de lógica empleando álgebra de Boole y teoría de proposiciones.

Indicador de logro

Aplica lógica matemática para resolver problemas.

1.1. Álgebra de Boole

El álgebra booleana (o Álgebra de Boole) proviene del trabajo realizado por el matemático de origen inglés llamado George Boole (1815 - 1864). El trabajo de George Boole ha trascendido en la historia, debido a que sus aportaciones permitieron reducir la lógica formal a un álgebra más práctica (dificultad que tenían las matemáticas en aquel entonces). Él logró una analogía con los símbolos algebraicos y las expresiones de sus formas lógicas en la Universidad de Cork, en Cork, Irlanda, centro de estudios booleanos. Allí, desde 1849 hasta su muerte en 1864, George Boole fue profesor de matemáticas (siendo el primero), nombrado en la fundación del Queen's College, en una época victoriana conocida como «la semilla del progreso moderno» [Crilly, 2025].

George Boole logró tanto en el ámbito matemático, gracias a una red social que lo apoyó. Su historia pone de manifiesto la importancia de los profesores a la hora de ayudar a los jóvenes matemáticos, así como la influencia que estos tienen. En el caso de George Boole, se aprecia el apoyo de dos personajes, la figura de su padre, y la nobleza obligada que le brindó el aristócrata local Sir Edward French Bromhead. Este hombre (Bromhead), que había pasado por el sistema *Tripas de Matemáticas de Cambridge*, pudo aconsejar a Boole sobre qué leer y estudiar en matemáticas.

George Boole tuvo acceso a maestros como Laplace y Lagrange, y esto se debió a la generosidad de Bromhead, que le permitió a su aprendiz utilizar su biblioteca en Thurlby Hall, tal y como había hecho veinte años antes con el igualmente pobre George Green, de Nottingham [Waszek, 2025]. Pero mientras Green pasó a la Universidad de Cambridge, Boole perdió esta oportunidad, muy a su pesar, aunque el sistema orientado a los exámenes que prevalecía entonces en Cambridge podría haber frenado su estilo natural. Las matemáticas en la época victoriana tenían una ventaja notable como objeto de estudio para los autodidactas de clase social más baja. No necesitaban costosos equipos científicos

ni un laboratorio, y era una materia en la que el talento y la dedicación implícitos podían superar las barreras de clase y económicas.

De forma particular, era posible, como demostró Boole, que siendo hijo de un zapatero se labrara una reputación científica. En esta línea, podemos citar otros ejemplos en el contexto británico, «estudiantes» que obtuvieron una beca de la Royal Society de Londres. Thomas P. Kirkman fue uno de ellos, elegido en la misma promoción que Boole en 1857. Anteriormente, el irlandés Robert Murphy, también hijo de un zapatero, fue elegido miembro en 1834 [Kyriazis, 2025].

Al principio de su carrera, George Boole recibió una medalla de oro por el trabajo matemático más significativo presentado a la Royal Society en el periodo comprendido entre junio de 1841 a junio de 1844, lo que sin duda supuso un estímulo vital para un joven, aunque el reconocimiento no le llegó fácilmente. George Boole tenía dos defensores. Thomas S. Davies, geómetra elegido miembro de la Royal Society en 1833, se ganaba la vida como tutor privado antes de ser nombrado profesor de matemáticas en la Real Academia Militar de Woolwich. Argumentó que *ser pobre no era una razón válida para que se rechazara su trabajo*. El otro apoyo vino de Escocia, en la persona de Philip Kelland, profesor de matemáticas en Edimburgo. Era un antiguo wrangler senior de Cambridge que podía comprender el valor del trabajo de Boole y cuyo juicio fue probablemente decisivo en la concesión de la medalla [Crilly, 2025].

George Boole publicó un libro sobre “*Las Leyes del pensamiento*”, presentando el sustento de las teorías de la lógica y la probabilidad, ver Fig. 1.1.

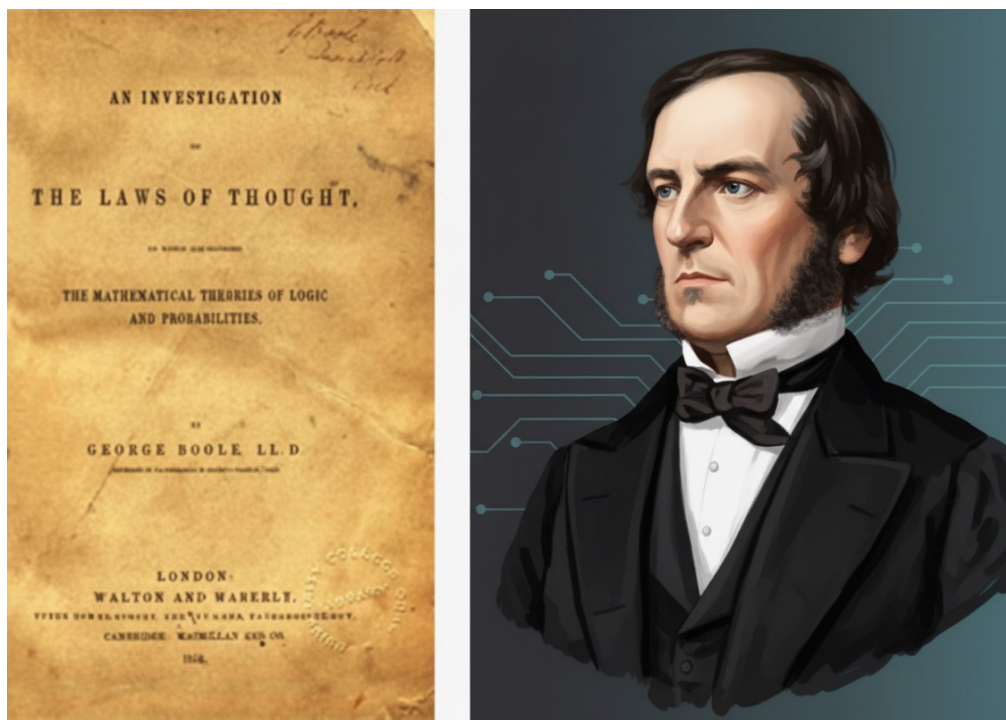


Figura 1.1: Imagen mejorada por la inteligencia artificial Gemini para la sección temática del Álgebra de Boole, en la imagen se muestra a George Boole y el tratado “Las leyes del pensamiento”

El álgebra booleana o de Boole, consiste en un método para resolver problemas de lógica, empleando para ello los valores binarios de 1 y 0 con el uso de tres operadores: AND (y), OR (o) y NOT (no), así como sus combinaciones.

1.1.1. Introducción

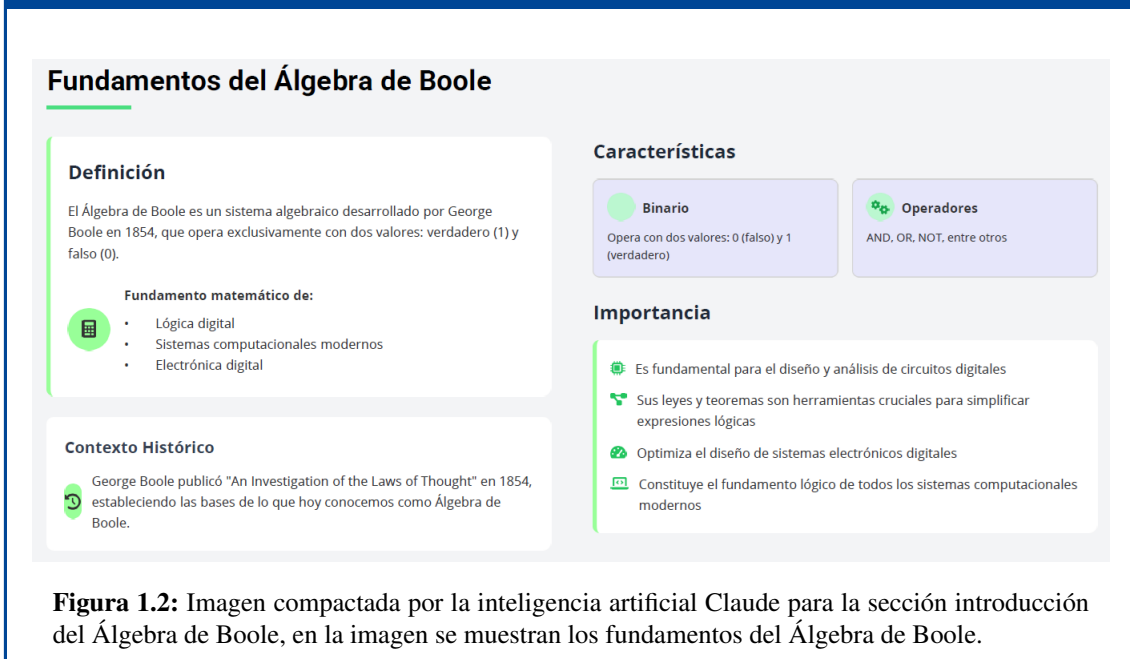
El álgebra booleana se lo utiliza en la creación de componentes electrónicos de las computadoras, placas y dispositivos. Fue desarrollada para manipular proposiciones lógicas de forma algebraica. Hoy es comprensible saber que los sistemas modernos trabajan con diseños de lógica binaria. Desde una condición inicial, el enfoque del álgebra booleana es abstracto, pero en 1938, la Bell Telephone Company en Estados Unidos la utilizó para analizar los circuitos de su red telefónica. Además, en ese mismo año, Claude Shannon creó el Álgebra de Conmutación a partir del álgebra booleana, lo que demostró que esta se adapta perfectamente al diseño y representación de circuitos lógicos de control.

Un sinnúmero de aplicaciones tiene el álgebra booleana y se muestran visibles en la implementación de los circuitos en los equipos de cómputo, en sistemas telefónicos, en la robótica y en cualquier operación automatizada de una empresa. En términos generales, el álgebra booleana se puede emplear en cualquier sistema en que cada variable posea dos estados (1 y 0).

- Se emplea en las computadoras modernas donde realizan sus operaciones y almacenan la información empleando valores binarios (1 y 0).
- Se emplea en el diseño de circuitos electrónicos para la elaboración de componentes eléctricos que ayudan a la conversión de lo analógico a lo digital (0V a 5V).
- Se emplea ampliamente en la programación en el momento de utilizar condiciones lógicas y validaciones que requieren una respuesta de un falso o un verdadero.

El álgebra booleana trabaja con señales binarias (un tipo de señal digital). A diferencia de una señal analógica (cambio continuo de magnitud), la señal binaria tiene solo dos valores posibles, conectado / desconectado o verdadero / falso (1 o 0) ver Fig. 1.2.

Para analizar y profundizar



1.1.2. Compuertas lógicas fundamentales

Las compuertas lógicas se constituyen en los elementos de construcción fundamental en la lógica combinatoria, con esto, debemos considerar lo siguiente, las compuertas trabajan con dos valores de “verdad”, los números, 1 y 0, y al igual que en la lógica proposicional, es posible la construcción de tablas de verdad para cada compuerta y, sus combinaciones con otras compuertas. Las variables de entrada se utilizan para representar el nivel de voltaje en un medio de cable eléctrico o en las entradas y salidas de un circuito electrónico (0V ó 5V).

Las compuertas son elementos de un circuito electrónico que se emplean para construir la mayoría de las expresiones lógicas formales (expresiones booleanas elementales). Existen tres compuertas principales: OR, AND y NOT. De éstas compuertas se derivan otras como son las compuertas NAND, NOR, XOR (OR exclusiva) y XNOR (NOR exclusiva) ver Fig. 1.3.



Figura 1.3: Imagen creada por la inteligencia artificial Gemini para la sección temática de compuertas lógicas fundamentales, en la imagen se muestra el imaginario de las compuertas lógicas.

Las compuertas lógicas, AND ($\wedge$) o Conjunción, OR ($\vee$) o Disyunción, y NOT ($\neg$) Negación, son utilizadas en sistemas digitales (base de los circuitos lógicos), circuitos electrónicos (diseño de sistemas de control automático), Bases de datos (consultas booleanas), Programación (condiciones lógicas y control de flujo) o Inteligencia artificial (sistemas de inferencia). En los sistemas digitales las variables booleanas representan niveles de voltaje, el número cero (o 0 lógico) representa al falso, apagado o bajo de voltaje (un valor en voltios de 0-0.8V), y el número uno (o un 1 lógico) representa al verdadero, encendido o alto de voltaje (un valor en voltios de 2-5V) y con esto se establece la ecuación, ver Tabla 1.1.

Compuertas lógicas y su fundamento del diseño digital

Tabla 1.1: Operaciones definidas de las compuertas lógicas.

Compuerta	Ecuación	Salida X	Tabla de Verdad
AND ($\wedge$)	$X = A \cdot B$	Salida 1 solo si todas las entradas son 1.	$0 \cdot 0 = 0, 0 \cdot 1 = 0, 1 \cdot 0 = 0, 1 \cdot 1 = 1$.
OR ($\vee$)	$X = A + B$	Salida 1 si al menos una entrada es 1.	$0 + 0 = 0, 0 + 1 = 1, 1 + 0 = 1, 1 + 1 = 1$.
NOT ($\neg$)	$X = \neg A = \bar{A}$	Salida es el inverso de la entrada.	$0 \rightarrow 1, 1 \rightarrow 0$.
NAND	$X = \overline{A \cdot B}$	Inverso de AND.	$0 \cdot 0 = 0 \rightarrow 1, 0 \cdot 1 = 0 \rightarrow 1, 1 \cdot 0 = 0 \rightarrow 1, 1 \cdot 1 = 1 \rightarrow 0$.
NOR	$X = \overline{A + B}$	Inverso de OR.	$0 + 0 = 0 \rightarrow 1, 0 + 1 = 1 \rightarrow 0, 1 + 0 = 1 \rightarrow 0, 1 + 1 = 1 \rightarrow 0$.
XOR ($\oplus$)	$X = A \cdot \bar{B} + \bar{A} \cdot B$	Salida 1 cuando las entradas son diferentes.	$00 \rightarrow 0, 01 \rightarrow 1, 10 \rightarrow 1, 11 \rightarrow 0$.
XNOR	$X = A \cdot B + \bar{A} \cdot \bar{B}$	Salida 1 cuando las entradas son iguales.	$00 \rightarrow 1, 01 \rightarrow 0, 10 \rightarrow 0, 11 \rightarrow 1$.

De acuerdo con estas operaciones, se formaliza una conexión entre los principios de la electrónica y la abstracción del Álgebra de Boole, que es de importancia para el diseño en la ingeniería. Comprender la estructura de las compuertas lógicas consiste en comprender el concepto físico del transistor y su representación abstracta como un artefacto conocido como interruptor lógico. El transistor actúa como un interruptor que al llegar energía o recibir una variación de voltaje alto permite o niega el flujo de corriente. Al llegar un voltaje alto, se representa un estado de verdad o un 1 lógico (circuito cerrado), mientras que la ausencia de voltaje, se representa un estado falso o un 0 lógico (circuito abierto).

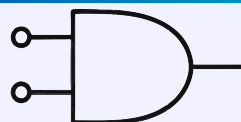
La **compuerta lógica AND** se la puede analizar desde diferentes ámbitos, como por ejemplo la multiplicación binaria o producto elemento a elemento (en álgebra lineal, al operar con vectores), la intersección (en teoría de conjuntos) o la conjunción (en lógica formal) (ver Fig. 1.4).

Compuerta lógica AND

La compuerta lógica AND realiza una multiplicación lógica, donde la salida es verdadera (1) únicamente cuando todas sus entradas son verdaderas (1).

Tabla de verdad AND

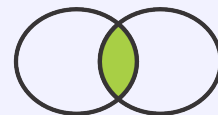
A	B	Operación	Salida
0	0	$0 \cdot 0$	0
0	1	$0 \cdot 1$	0
1	0	$1 \cdot 0$	0
1	1	$1 \cdot 1$	1



$$X = A \cdot B$$

AND o Y

Multiplicación de Componentes ($\cdot$)
Intersección ($\cap$)
Conjunción ($\wedge$)



En los lenguajes de programación se lo define como el ampersand & (uno) o && (dos seguidos).

Figura 1.4: Descripción de la información de la compuerta lógica AND

La compuerta lógica OR (OR Inclusiva u O inclusiva), cuya salida es 1 si al menos una entrada es 1, tiene analogías en diversas áreas de las matemáticas y la lógica. En la *Lógica Formal*, corresponde a la *disyunción* ($\vee$). En *Teoría de Conjuntos*, el operador OR es equivalente a la *Unión* ($\cup$) de conjuntos, recuerda que la unión de dos conjuntos $A \cup B$ incluye elementos que están en A o en B . Finalmente, en el *Álgebra Binaria y Lineal*, la compuerta lógica OR se modela como una *suma binaria con saturación*, donde el resultado de $1 + 1$ es forzado a 1 (a menudo representado como $\max(a, b)$ en operaciones componente a componente con vectores) (ver Fig. 1.5).

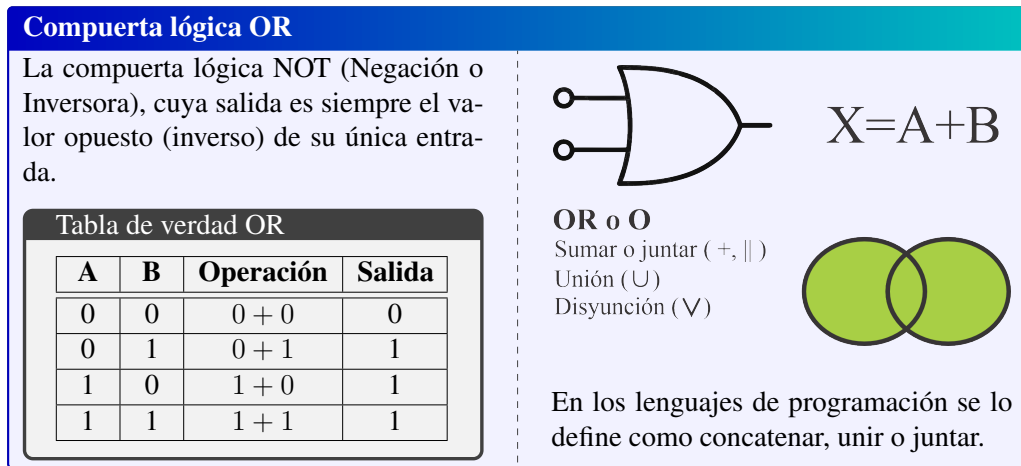


Figura 1.5: Descripción de la información de la compuerta lógica OR

La compuerta lógica NOT, NOT (Negación o Inversora), cuya salida es el valor opuesto de su única entrada, tiene analogías en diversas áreas de las matemáticas y la lógica. En la *Lógica Formal*, se corresponde con la *Negación* ($\neg$), donde si una proposición es verdadera, su negación es falsa, y viceversa. En *Teoría de Conjuntos*, el operador NOT es equivalente al *Complemento* ($\bar{A}$), representando todos los elementos que no pertenecen al conjunto A . Finalmente, en el *Álgebra Binaria y Aritmética*, la compuerta NOT se modela como la *resta de la unidad* ($1 - A$), realizando la inversión del valor (ver Fig. 1.6).

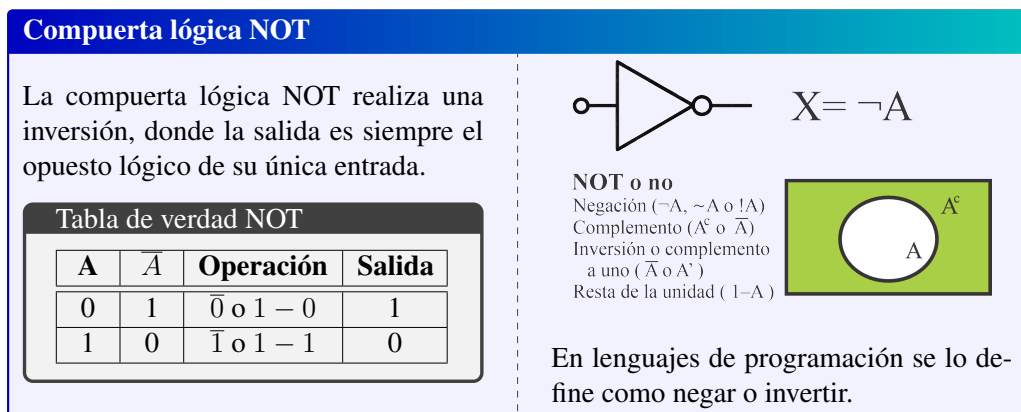


Figura 1.6: Descripción de la información de la compuerta lógica NOT

La compuerta lógica NAND (o AND negado), cuya salida es el inverso lógico de la compuerta AND, la salida es 0 solo cuando todas las entradas son 1. Se la puede analizar desde diferentes ámbitos, como por ejemplo la *negación del producto lógico* o *resta de la unidad al producto* (en álgebra binaria y lineal), la *negación de la intersección* (en teoría de conjuntos) o la *incompatibilidad* (en lógica formal) (ver Fig. 1.7).

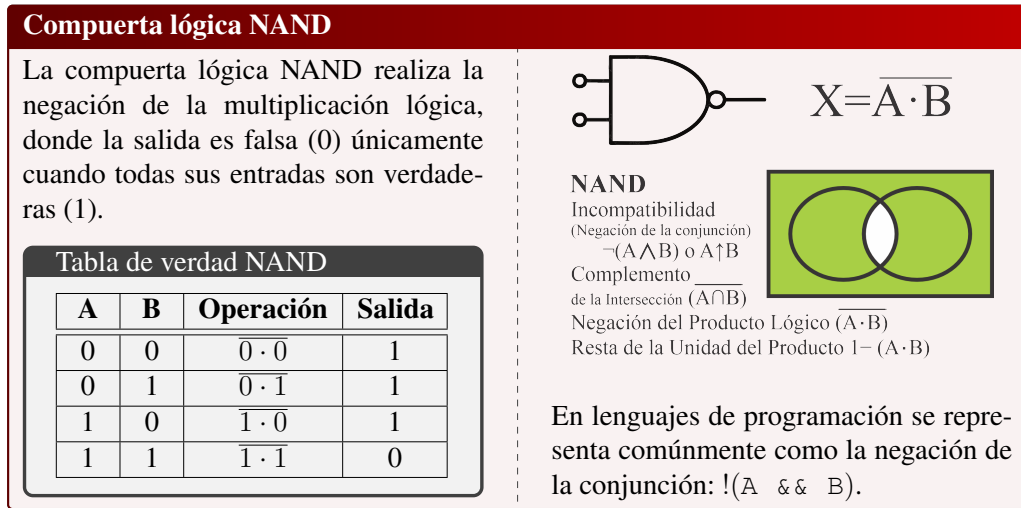


Figura 1.7: Descripción de la información de la compuerta lógica NAND

La compuerta lógica NOR (negación de la compuerta lógica OR), cuya salida es el inverso lógico de la compuerta OR, se puede analizar desde diferentes ámbitos, como por ejemplo la *negación de la suma lógica* o *resta de la unidad a la suma saturada* (en álgebra binaria y lineal), el *complemento de la unión* (en teoría de conjuntos) o la *negación de la disyunción* (en lógica formal) (ver Fig. 1.8).

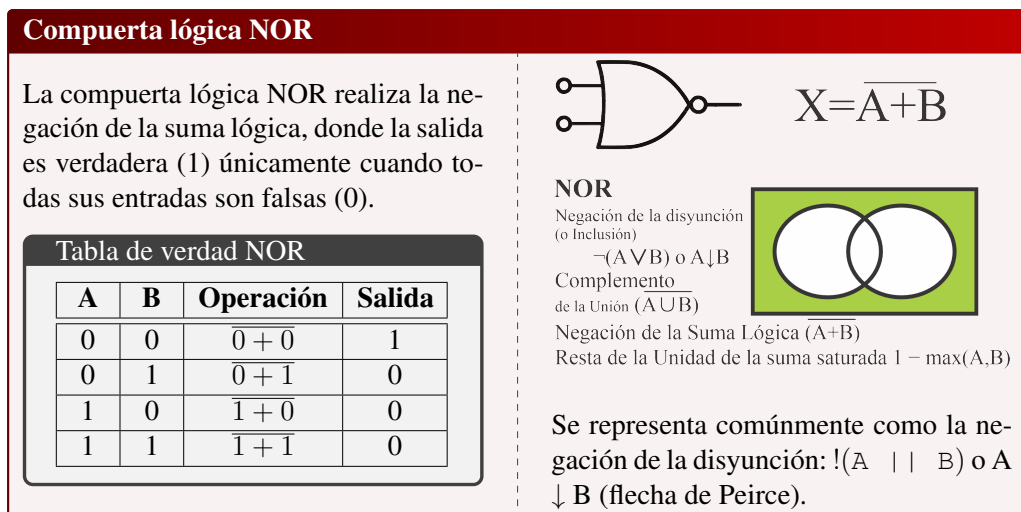


Figura 1.8: Descripción de la información de la compuerta lógica NOR

La compuerta lógica XOR (OR Exclusiva u O exclusiva), cuya salida es 1 si y solo si las entradas son diferentes, se puede analizar desde diferentes ámbitos, como por ejemplo la *suma módulo 2* o *detector de diferencia* (en álgebra binaria y lineal), la *diferencia simétrica* (en teoría de conjuntos) o la *disyunción exclusiva* (en lógica formal) (ver Fig. 1.9).

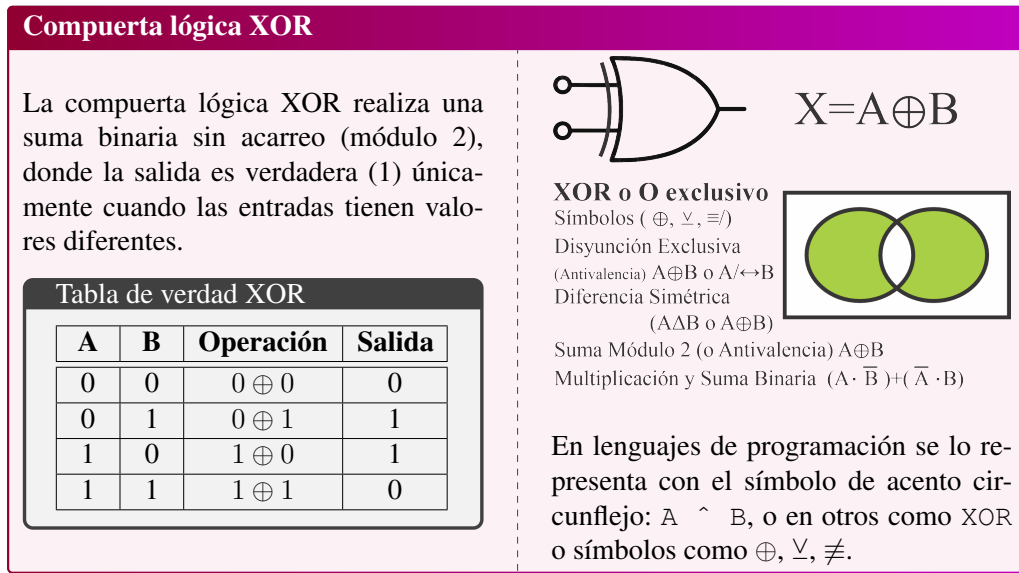


Figura 1.9: Descripción de la información de la compuerta lógica XOR

La compuerta lógica XNOR (OR Exclusiva Negada o Equivalencia), cuya salida es 1 si y solo si las entradas son iguales, se puede analizar desde diferentes ámbitos, como por ejemplo la *coincidencia* o *negación de la suma módulo 2* (en álgebra binaria y lineal), el *complemento de la diferencia simétrica* (en teoría de conjuntos) o la *equivalencia* o *bicondicional* (en lógica formal) (ver Fig. 1.10).

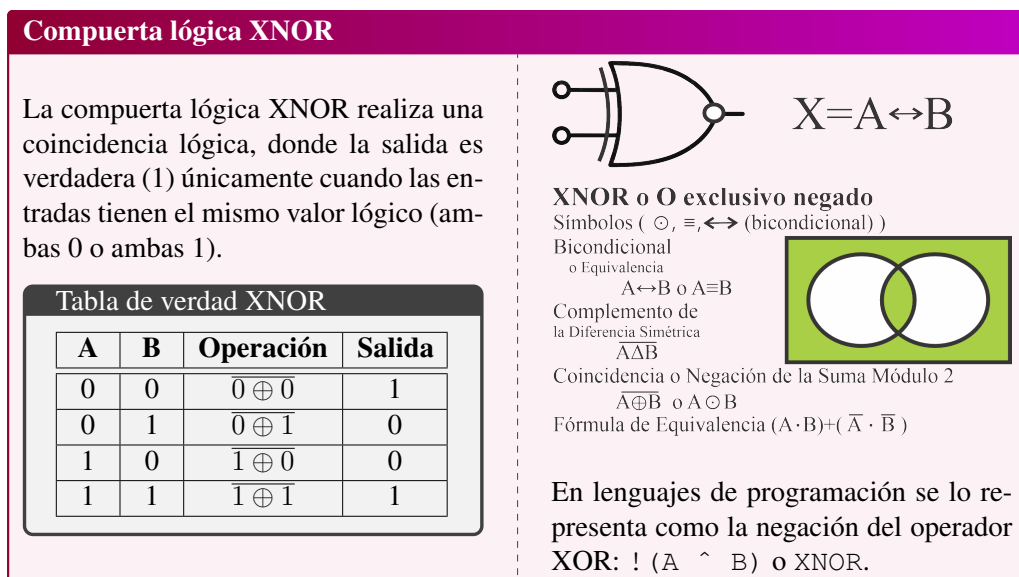


Figura 1.10: Descripción de la información de la compuerta lógica XNOR

1.1.3. Leyes y teoremas

George Boole nació en una familia de clase baja, donde su padre siendo zapatero le inculcó la motivación por las matemáticas y su aplicación a la construcción de dispositivos ópticos e instrumentos científicos. Sus padres fueron John Boole y Mary Ann Joyce, y siendo el primero de cuatro hermanos, fue en gran medida un autodidacta¹. Sin una educación formal, tuvo que abordar los problemas de manera un poco convencional. Tuvo que elegir nuevos enfoques de estudio en vez de seguir los métodos tradicionales. Buscó la verdad en la analogía entre el álgebra numérica y la lógica simbólica, ver Fig. 1.11.

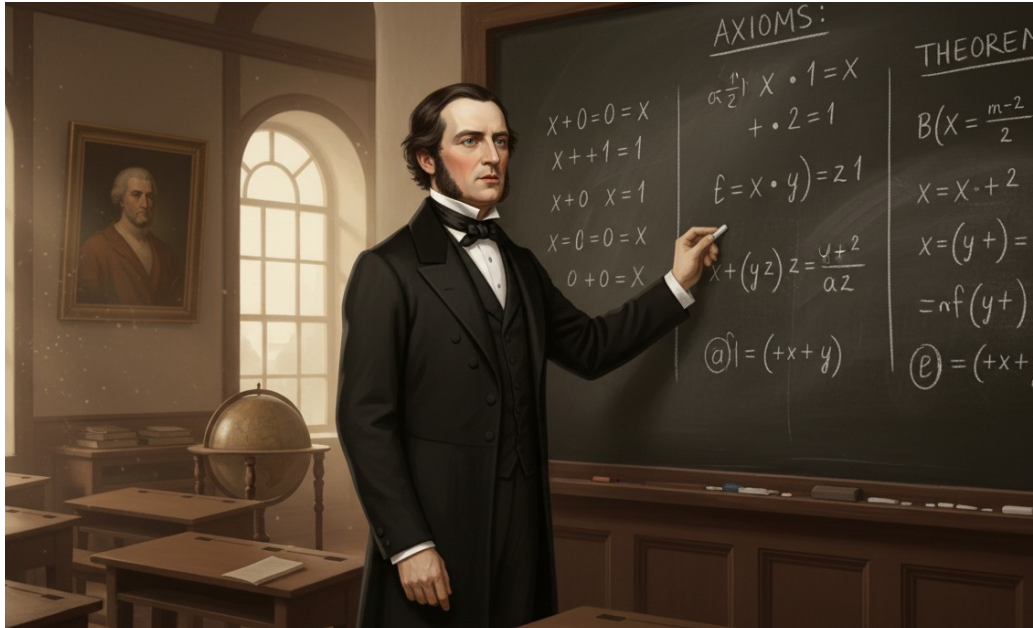


Figura 1.11: Imagen creada por la inteligencia artificial Gemini para la sección temática de leyes y teoremas, en la imagen se muestra imaginariamente a George Boole y los axiomas.

A partir de esta situación en su vida, su inventiva y genialidad no se limitó por la falta de un sistema educativo tradicional, sino que a su vez, se desarrolló a través de la exploración independiente y la disciplina personal. Ya en el siglo XIX, la manera de interpretar la lógica era mediante la filosofía. George Boole entendió que la lógica podía ser desarrollada e interpretada con métodos algebraicos.

Con esto escribió su obra “*Análisis Matemático de la Lógica*” (1847), un breve folleto en el que demostró que la lógica debía ser estudiada como una rama de las matemáticas. Esta idea evidenció su profundo pensamiento innovador sobre la universalidad de las estructuras matemáticas, que podían aplicarse a espacios que tradicionalmente se consideraban ajenos a ellas. George Boole estaba seguro de que la mente del ser humano opera siguiendo un conjunto de leyes formales, y su álgebra booleana era un intento de decodificar, precisar y formalizar esas “*leyes del pensamiento*”.

El álgebra booleana se enfoca al establecimiento de las bases de un proceso matemático teórico completo que permite describir la lógica binaria. Así, un axioma llega a ser un principio claro o una proposición que se considera evidente por sí misma sin necesidad de una demostración. Es la estructura conceptual base para una teoría, área de estudio o

¹[Biografía de George Boole](#)

diseño de un sistema lógico. Para George Boole el principio de la dualidad establece que a partir de cualquier expresión booleana o ecuación de identidad mantiene su validez aun si se intercambian los operadores de la disyunción (OR o +) o de la conjunción (AND o $\cdot$), ver Tabla 1.2.

Leyes Fundamentales del Álgebra de Boole		
Tabla 1.2: Axiomas y teoremas esenciales para la simplificación lógica.		
Leyes	Operación Suma (+)	Operación Producto ($\cdot$)
Identidad	$A + 0 = A$	$A \cdot 1 = A$
Complemento	$A + \bar{A} = 1$	$A \cdot \bar{A} = 0$
Dominación	$A + 1 = 1$	$A \cdot 0 = 0$
Idempotencia	$A + A = A$	$A \cdot A = A$
Conmutativas	$A + B = B + A$	$A \cdot B = B \cdot A$
Asociativas	$A + (B + C) = (A + B) + C$	$A \cdot (B \cdot C) = (A \cdot B) \cdot C$
Distributivas	$A \cdot (B + C) = A \cdot B + A \cdot C$	$A + (B \cdot C) = (A + B) \cdot (A + C)$
De Morgan	$(A + B)' = \bar{A} \cdot \bar{B}$	$(A \cdot B)' = \bar{A} + \bar{B}$
Doble Negación	$(\bar{\bar{A}}) = A$	

George Boole observó que los símbolos algebraicos para números seguían patrones que obedecen a ciertas leyes, como $x^2 = x$, una ley que en el álgebra numérica solo se cumple para el cero o el uno. Esta observación fue el punto importante para reducir las expresiones lógicas a un cálculo de solo dos valores. Él interpretó 1 como el “*universo*” (lo que contiene todo) y 0 como la “*nada*”. Esta intuición, que conectaba una propiedad algebraica numérica con la naturaleza binaria de la lógica, es un ejemplo significativo de su capacidad para ver conexiones profundas entre disciplinas aparentemente diferentes, ver Fig. 1.12.



Figura 1.12: Imagen creada por la inteligencia artificial Gemini para la sección temática de leyes y teoremas, en la imagen se muestra imaginariamente a George Boole y un contemporáneo.

Él sentó las bases para el concepto de bit en la era digital. Por eso los axiomas definen el comportamiento de las operaciones fundamentales anteriormente conocidas (OR, AND y NOT). Estos principios y las reglas de inferencia, representan el sistema axiomático a partir del cual, George Boole, deriva todos sus demás teoremas.

La importancia del trabajo de George Boole tuvo relevancia muchos años después de su muerte. Fue Claude Shannon, en 1938, quien aplicó el álgebra booleana a los circuitos de conmutación electrónica, lo que sentó las bases de la electrónica digital y la informática moderna. Él no estaba simplemente resolviendo un problema de su tiempo, estaba construyendo un marco teórico para un futuro tecnológico que aún no existía. Su pensamiento era tan fundamental y muy adelantado a su tiempo que su verdadero impacto se hizo evidente muchos años después. Su trabajo es un testimonio de vida y de la importancia de la investigación teórica fundamental, incluso cuando sus aplicaciones prácticas no son inmediatamente visibles en su tiempo.

1.1.4. Simplificación de expresiones booleanas

George Boole transformó la lógica (a partir de la filosofía), utilizando un sistema de expresiones algebraicas, permitiendo un análisis matemático formal de las proposiciones lógicas. Este desarrollo, el álgebra booleana, se convirtió en un sistema base de la informática moderna. A pesar de la admiración póstuma, sus contemporáneos no entendieron su trabajo revolucionario, que aplicaba a problemas matemáticos como el análisis de ecuaciones diferenciales y métodos de probabilidad, ver Fig. 1.13.



Figura 1.13: Imagen creada por la inteligencia artificial Gemini para la sección temática de simplificación de expresiones booleanas, en la imagen se muestran esculturas póstumas.

Boole cambió la forma de tratar la lógica, su artículo “*Sobre un método general de análisis*”, aplicó métodos algebraicos a las ecuaciones diferenciales, lo que le valió la Medalla Real de Matemáticas en 1844, la primera otorgada a un matemático. Este reconocimiento lo llevó a conseguir una cátedra en el Queen’s College Cork. Boole no vivió

para ver el impacto de su obra. Décadas más tarde, Claude Shannon reconoció la aplicabilidad del álgebra booleana al diseño de circuitos eléctricos, sentando las bases para la era digital.

A partir de la aplicación de las leyes, teoremas y axiomas, George Boole propuso una forma de procesar las expresiones lógicas de tal manera que estas se puedan reducir a una forma más sencilla o equivalente, y con un número menor de términos o variables. La simplificación lógica se puede utilizar en dos ámbitos, en el diseño y fabricación de placas con circuitos digitales y en la programación.

En el primer ámbito, al diseñar circuitos con menos compuertas y con la misma función, se traduce en la disminución del costo de fabricación, el menor consumo de energía y en la reducción de la complejidad del diseño. En el segundo ámbito, al realizar programas, la simplificación permite reducir expresiones lógicas muy complejas en el código, a versiones más pequeñas y legibles, con buen rendimiento y eficientes. Existen varias formas para realizar la simplificación lógica, entre ellas están, el álgebra booleana y sus ecuaciones de equivalencia (reglas), mapas de Karnaugh (K-map o método gráfico) o el algoritmo de Quine-McCluskey (o método tabular), ver Tabla 1.3.

Leyes y teoremas fundamentales del Álgebra de Boole

Tabla 1.3: Compendio de las leyes y teoremas esenciales para la simplificación lógica.

Categoría / Ley	Operación Suma (OR: +)	Operación Producto (AND: ·)
I. Leyes axiomáticas y básicas		
Identidad	$A + 0 = A$	$A \cdot 1 = A$
Complemento (Inversión)	$A + \overline{A} = 1 \equiv A + A' = 1$	$A \cdot \overline{A} = 0 \equiv A \cdot A' = 0$
Dominación (Anulación)	$A + 1 = 1$	$A \cdot 0 = 0$
Idempotencia	$A + A = A$	$A \cdot A = A$
II. Leyes estructuras y negación		
Conmutativas	$A + B = B + A$	$A \cdot B = B \cdot A$
Asociativas	$A + (B + C) = (A + B) + C$	$A \cdot (B \cdot C) = (A \cdot B) \cdot C$
Distributivas	$A \cdot (B + C) = A \cdot B + A \cdot C$	$A + (B \cdot C) = (A + B) \cdot (A + C)$
Doble Negación	$\overline{\overline{A}} = A$, equivale escribir a $(A')' = A \equiv A'' = A$	
Negación de Constantes	$\overline{1} = 0$	$\overline{0} = 1$
Unicidad del Complemento	Si $A + X = 1$ y $A \cdot X = 0$, entonces $X = \overline{A}$	
III. Teoremas de simplificación		
De Morgan	$(A + B)' = \overline{A} \cdot \overline{B}$	$(A \cdot B)' = \overline{A} + \overline{B}$
Absorción	$A + A \cdot B = \mathbf{A}$	
Desarrollo	$A + A \cdot B = A \cdot (1 + B) = A \cdot 1 = \mathbf{A}$	
Absorción Dual	$A \cdot (A + B) = \mathbf{A}$	
Desarrollo	$A \cdot (A + B) = A \cdot A + A \cdot B = A + A \cdot B = \mathbf{A}$	
Simplificación Binomial (Unificación)	$A \cdot B + A \cdot B' = \mathbf{A}$	
Desarrollo	$A \cdot B + A \cdot B' = A \cdot (B + B') = A \cdot 1 = \mathbf{A}$	
Simplificación Binomial Dual	$(A + B) \cdot (A + B') = \mathbf{A}$	
Desarrollo	$(A + B) \cdot (A + B') = A + B \cdot B' = A + 0 = \mathbf{A}$	
Consenso (Eliminación)	$A \cdot B + A' \cdot C + B \cdot C = A \cdot B + A' \cdot C$	
Término Eliminado	El término $B \cdot C$ es el consenso redundante de $A \cdot B$ y $A' \cdot C$.	

A continuación, se presenta un ejemplo de simplificación de una expresión booleana

siendo su expresión original:

$$F = ABC + ABC' + A'BC$$

Entonces su desarrollo y justificación sería el siguiente:

Paso 1: Factorizar AB en los dos primeros términos.

$$F = AB(C + C') + A'BC \quad (\text{Factorización})$$

Paso 2: Aplicar la Ley del Complemento ($X + X' = 1$).

$$F = AB(1) + A'BC \quad (\text{Ley del Complemento: } C + C' = 1)$$

Paso 3: Aplicar la Ley de Identidad ($X \cdot 1 = X$).

$$F = AB + A'BC \quad (\text{Ley de Identidad})$$

Paso 4: Factorizar B .

$$F = B(A + A'C) \quad (\text{Factorización})$$

Paso 5: Aplicar el Teorema de Absorción Extendido ($X + X'Y = X + Y$).

$$F = B(A + C) \quad (\text{Teorema de Absorción Extendido: } A + A'C = A + C)$$

Resultado simplificado:

$$\mathbf{F = B(A + C) = AB + BC}$$

A continuación, se presenta otro ejemplo de simplificación de una expresión booleana siendo su expresión original:

$$F = (A + B')(A + B)C' + A'$$

Entonces su desarrollo y justificación sería el siguiente (ver Tabla 1.4):

Ejemplo de simplificación de expresiones booleanas (Método algebraico)

Tabla 1.4: Simplificación de $F = (A + \overline{B})(A + B)\overline{C} + \overline{A}$

Paso	Expresión Resultante	Ley o Teorema Aplicado
Inicial	$F = (A + \overline{B})(A + B)\overline{C} + \overline{A}$	Expresión Original
1	$F = [A + (B \cdot B)]\overline{C} + \overline{A}$	Ley Distributiva inversa: $(A + X)(A + Y) = A + (X \cdot Y)$
2	$F = [A + 0]\overline{C} + \overline{A}$	Ley de Complemento (AND): $B \cdot \overline{B} = 0$
3	$F = A \cdot \overline{C} + \overline{A}$	Ley de Identidad (OR): $A + 0 = A$
4	$F = \overline{A} + A\overline{C}$	Ley Conmutativa (OR): $X + Y = Y + X$
5	$F = (\overline{A} + A)(\overline{A} + \overline{C})$	Ley Distributiva: $X + (Y \cdot Z) = (X + Y)(X + Z)$
6	$F = (1)(\overline{A} + \overline{C})$	Ley de Complemento (OR): $\overline{A} + A = 1$
7	$F = \overline{A} + \overline{C}$	Ley de Identidad (AND): $1 \cdot X = X$

Resultado final simplificado:

$$F = \overline{A} + \overline{C}$$

A continuación, se presenta otro ejemplo de simplificación de una expresión booleana, utilizando las leyes y teoremas fundamentales, siendo la expresión original es: $F = \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC$, entonces su desarrollo y justificación sería el siguiente (ver Tabla 1.5):

Ejemplo de simplificación de expresiones booleanas (Método algebraico)

Tabla 1.5: Simplificación de $F = \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC$

Paso	Expresión Resultante	Ley o Teorema Aplicado
Inicial	$F = \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC$	Expresión Original
1	$F = (\overline{A}BC + ABC) + (A\overline{B}C + ABC) + (AB\overline{C} + ABC)$	Ley de Idempotencia: ABC se duplica para agrupar (Consenso).
2	$F = (\overline{A} + A)BC + (\overline{B} + B)AC + (\overline{C} + C)AB$	Ley Distributiva: Factor común en cada par de términos.
3	$F = (1)BC + (1)AC + (1)AB$	Ley de Complemento (OR): $X + \overline{X} = 1$
4	$F = BC + AC + AB$	Ley de Identidad (AND): $X \cdot 1 = X$

Resultado final simplificado:

$$F = AC + AB + BC$$

A continuación, se presenta otro ejemplo de simplificación de una expresión booleana, utilizando las leyes y teoremas fundamentales, siendo la expresión original es: $F = AB\overline{C} + ABC + AB\overline{D} + \overline{A}BC$, entonces su desarrollo y justificación sería el siguiente (ver Tabla 1.6):

Ejemplo de simplificación de expresiones booleanas (Método algebraico)

Tabla 1.6: Simplificación de $F = AB\overline{C} + ABC + AB\overline{D} + \overline{A}BC$

Paso	Expresión Resultante	Ley o Teorema Aplicado
Inicial	$F = AB\overline{C} + ABC + AB\overline{D} + \overline{A}BC$	Expresión Original
1	$F = AB(\overline{C} + C) + AB\overline{D} + \overline{A}BC$	Ley Distributiva: Factor común AB en los primeros dos términos.
2	$F = AB(1) + AB\overline{D} + \overline{A}BC$	Ley de Complemento (OR): $X + \overline{X} = 1$
3	$F = AB + AB\overline{D} + \overline{A}BC$	Ley de Identidad (AND): $X \cdot 1 = X$
4	$F = AB(1 + \overline{D}) + \overline{A}BC$	Ley Distributiva: Factor común AB en los dos primeros términos.
5	$F = AB(1) + \overline{A}BC$	Ley de Dominación (OR): $1 + X = 1$
6	$F = AB + \overline{A}BC$	Ley de Identidad (AND): $X \cdot 1 = X$

Resultado final simplificado:

$$F = AB + \overline{A}BC$$

A continuación, se presenta otro ejemplo de simplificación de una expresión booleana, utilizando las leyes y teoremas fundamentales, siendo la expresión original: $F = \overline{A}B + AB + \overline{A}BC$, entonces su desarrollo y justificación sería el siguiente (ver Tabla 1.7):

Ejemplo de simplificación de expresiones booleanas (Método algebraico)

Tabla 1.7: Simplificación de $F = \overline{A}B + AB + \overline{A}BC$

Paso	Expresión Resultante	Ley o Teorema Aplicado
Inicial	$F = \overline{A}B + AB + \overline{A}BC$	Expresión Original (Reordenada por Conmutativa)
1	$F = B(\overline{A} + A) + \overline{A}BC$	Ley Distributiva: Factor común B en los primeros dos términos.
2	$F = B(1) + \overline{A}BC$	Ley de Complemento (OR): $\overline{A} + A = 1$
3	$F = B + \overline{A}BC$	Ley de Identidad (AND): $B \cdot 1 = B$
4	$F = B + \overline{A}C$	Teorema de Absorción: $X + \overline{X}Y = X + Y$. Aquí, $X = B$ y $Y = \overline{A}C$.

Resultado final simplificado:

$$F = B + \overline{A}C$$

A continuación, se presenta otro ejemplo de simplificación de una expresión booleana, utilizando las leyes y teoremas fundamentales, siendo la expresión original: $F = A(\overline{A} + B) + B(\overline{B} + C)$, entonces su desarrollo y justificación sería el siguiente (ver Tabla 1.8):

Ejemplo de simplificación de expresiones booleanas (Método algebraico)

Tabla 1.8: Simplificación de $F = A(\overline{A} + B) + B(\overline{B} + C)$

Paso	Expresión Resultante	Ley o Teorema Aplicado
Inicial	$F = A(\overline{A} + B) + B(\overline{B} + C)$	Expresión Original
1	$F = A\overline{A} + AB + B\overline{B} + BC$	Ley Distributiva: $X(Y + Z) = XY + XZ$
2	$F = 0 + AB + 0 + BC$	Ley de Complemento (AND): $X\overline{X} = 0$
3	$F = AB + BC$	Ley de Identidad (OR): $X + 0 = X$
4 (Alternativa)	$F = B(A + C)$	Ley Distributiva (Factorización): $XY + XZ = X(Y + Z)$

Resultado final simplificado:

$$F = B(A + C)$$

A continuación, se presenta otro ejemplo de simplificación de una expresión booleana, utilizando las leyes y teoremas fundamentales, siendo la expresión original: $F = AB + A\bar{B} + \bar{A}C$, entonces su desarrollo y justificación sería el siguiente (ver Tabla 1.9):

Ejemplo de simplificación de expresiones booleanas (Método algebraico)

Tabla 1.9: Simplificación de $F = AB + A\bar{B} + \bar{A}C$

Paso	Expresión Resultante	Ley o Teorema Aplicado
Inicial	$F = AB + A\bar{B} + \bar{A}C$	Expresión Original (Reordenada por Conmutativa si se desea)
1	$F = A(B + \bar{B}) + \bar{A}C$	Ley Distributiva: Factor común A en los dos primeros términos.
2	$F = A(1) + \bar{A}C$	Ley de Complemento (OR): $B + \bar{B} = 1$
3	$F = A + \bar{A}C$	Ley de Identidad (AND): $A \cdot 1 = A$
4	$F = A + C$	Teorema de Absorción: $X + \bar{X}Y = X + Y$. Aquí, $X = A$ y $Y = C$.

Resultado final simplificado:

$$F = A + C$$

A continuación, se presenta otro ejemplo de simplificación de una expresión booleana, utilizando las leyes y teoremas fundamentales, siendo la expresión original: $F = AC + A\bar{C} + BC$, entonces su desarrollo y justificación sería el siguiente (ver Tabla 1.10):

Ejemplo de simplificación de expresiones booleanas (Método algebraico)

Tabla 1.10: Simplificación de $F = AC + A\bar{C} + BC$

Paso	Expresión Resultante	Ley o Teorema Aplicado
Inicial	$F = AC + A\bar{C} + BC$	Expresión Original (Reordenada por Conmutativa)
1	$F = A(C + \bar{C}) + BC$	Ley Distributiva: Factor común A en los dos primeros términos.
2	$F = A(1) + BC$	Ley de Complemento (OR): $C + \bar{C} = 1$
3	$F = A + BC$	Ley de Identidad (AND): $A \cdot 1 = A$
4	$F = A + C$	Teorema de Absorción: $X + YZ = (X + Y)(X + Z)$. Con $X = A$, $Y = B$, $Z = C$, se obtiene $A + BC = (A + B)(A + C)$ y simplificando: $F = A + C$.

Resultado final simplificado:

$$F = A + C$$

A continuación, se presenta otro ejemplo de simplificación de una expresión booleana con cuatro variables, utilizando las leyes y teoremas fundamentales, siendo la expresión original: $(F = ABC + ABC + \overline{A}BD + \overline{A}BD)$, entonces su desarrollo y justificación sería el siguiente (ver Tabla 1.11):

Ejemplo de simplificación de expresiones booleanas (4 variables, método algebraico)

Tabla 1.11: Simplificación de $F = ABC + ABC + \overline{A}BD + \overline{A}BD$

Paso	Expresión Resultante	Ley o Teorema Aplicado
Inicial	$F = ABC + ABC + \overline{A}BD + \overline{A}BD$	Expresión Original (ordenada para agrupar términos semejantes)
1	$F = AB(C + \overline{C}) + D(\overline{A}B + \overline{A}B)$	Ley Distributiva: factor común AB en los dos primeros términos y factor común D en los dos últimos.
2	$F = AB(1) + D(\overline{B}(A + \overline{A}))$	Reagrupación y Distributiva: extraer $\overline{B}$ en el segundo grupo.
3	$F = AB + D(\overline{B} \cdot 1)$	Ley de Complemento: $C + \overline{C} = 1$ y $A + \overline{A} = 1$.
4	$F = AB + D\overline{B}$	Ley de Identidad (AND): $\overline{B} \cdot 1 = \overline{B}$.

Resultado final simplificado:

$$F = AB + \overline{B}D$$

Además se presenta la tabla de verdad resultante (ver Tabla 1.12):

Tabla de verdad de $F = AB + \overline{B}D$

Tabla 1.12: Tabla de verdad de $F(A, B, C, D)$ con $F = AB + \overline{B}D$

A	B	C	D	F
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

Luego se procede a realizar el proceso de creación del mapa de Karnaugh del ejercicio.

Mapa de Karnaugh (4 variables) para F

AB	CD			
	00	01	11	10
00	0	1	1	0
01	0	0	0	0
11	1	1	1	1
10	0	1	1	0

En el mapa se identifican dos grupos mínimos:

- Toda la fila $AB = 11$, que corresponde al término AB .
- Un grupo de cuatro celdas con $B = 0$ y $D = 1$, que corresponde a $\overline{B}D$.

Por tanto la forma mínima es:

$$F = AB + \overline{B}D.$$

Nota sobre el consenso: El término de consenso entre AB y $\overline{B}D$ es AD . Sin embargo, este no reduce la expresión porque es redundante:

$$AB + \overline{B}D + AD = AB + \overline{B}D.$$

Por tanto la forma obtenida ya es mínima.

Luego se procede a generar el mapa de Karnaugh del ejercicio.

Mapa de Karnaugh con agrupamientos ($F = AB + \overline{B}D$)

	00	01	11	10
00	0	1	1	0
01	0	0	0	0
11	1	1	1	1
10	0	1	1	0

Resultado de la simplificación:

$$F = AB + \overline{B}D$$

A continuación, se presenta otro ejemplo siendo la expresión original: $F = AB + \overline{A}\overline{B}C + \overline{A}BC + \overline{A}\overline{B}C$, entonces su desarrollo sería el siguiente (ver Tabla 1.13):

Ejemplo de simplificación de expresiones booleanas (Método algebraico)

Tabla 1.13: Simplificación de $F = AB + \overline{A}\overline{B}C + \overline{A}BC + \overline{A}\overline{B}C$

Paso	Expresión Resultante	Ley o Teorema Aplicado
Inicial	$F = AB + \overline{A}\overline{B}C + \overline{A}BC + \overline{A}\overline{B}C$	Expresión Original
1	$F = AB + \overline{A}\overline{B}C + C(\overline{A}\overline{B} + \overline{A}B)$	Ley Distributiva: Factor común C en los últimos dos términos.
2	$F = AB + \overline{A}\overline{B}C + C\overline{A}(B + \overline{B})$	Ley Distributiva: Factor común $\overline{A}$ dentro del paréntesis.
3	$F = AB + \overline{A}\overline{B}C + C\overline{A}(1)$	Ley de Complemento (OR): $B + \overline{B} = 1$
4	$F = AB + \overline{A}\overline{B}C + \overline{A}C$	Ley de Identidad (AND): $C\overline{A} \cdot 1 = \overline{A}C$
5	$F = AB + \overline{A}C + \overline{A}\overline{B}C$	Ley Conmutativa: Reordenamiento de términos.
6	$F = AB + C(\overline{A} + \overline{A}\overline{B})$	Ley Distributiva: Factor común C en los últimos dos términos.
7	$F = AB + C(\overline{A} + \overline{B})$	Teorema de Absorción: $X + XY = X$. Aquí, $\overline{A} + \overline{A}\overline{B} = \overline{A} + \overline{B}$
8	$F = AB + C\overline{A}\overline{B}$	Ley de De Morgan: $\overline{A} + \overline{B} = \overline{A\overline{B}}$

Resultado final simplificado:

$$F = AB + C\overline{A}\overline{B}$$

En cambio, en este ejemplo se muestra la simplificación paso a paso de una expresión booleana de 4 términos, aplicando sistemáticamente las leyes distributiva, de complemento, de identidad, de absorción y de De Morgan para llegar a una forma más compacta con solo 2 términos. Luego generamos el mapa de Karnaugh del ejercicio.

Mapa de Karnaugh con agrupamientos ($F = AB + C\overline{A}\overline{B}$)

	0	1
00	0	1
01	0	1
11	1	1
10	0	1

La simplificación $F = AB + C\overline{A}\overline{B}$, tiene **agrupamientos** en **azul**, AB (fila $AB=11$, que cubre ambos valores de C) y en **verde**, $C\overline{A}\overline{B}$ (columna $C=1$, filas $AB=00, 01, 10$)

A continuación, se presenta otro ejemplo siendo la expresión original: $F = \overline{A}BD + ABD + \overline{A}\overline{B}D + \overline{A}\overline{B}\overline{C}$, entonces su desarrollo sería el siguiente (ver Tabla 1.14):

Ejemplo de simplificación de expresiones booleanas (Método algebraico)

Tabla 1.14: Simplificación de $F = \overline{A}BD + ABD + \overline{A}\overline{B}D + \overline{A}\overline{B}\overline{C}$

Paso	Expresión Resultante	Ley o Teorema Aplicado
Inicial	$F = \overline{A}BD + ABD + \overline{A}\overline{B}D + \overline{A}\overline{B}\overline{C}$	Expresión Original
1	$F = BD(\overline{A} + A) + \overline{A}\overline{B}D + \overline{A}\overline{B}\overline{C}$	Ley Distributiva: Factor común BD en los primeros dos términos.
2	$F = BD(1) + \overline{A}\overline{B}D + \overline{A}\overline{B}\overline{C}$	Ley de Complemento (OR): $\overline{A} + A = 1$
3	$F = BD + \overline{A}\overline{B}D + \overline{A}\overline{B}\overline{C}$	Ley de Identidad (AND): $BD \cdot 1 = BD$
4	$F = BD + \overline{A}\overline{B}(D + \overline{C})$	Ley Distributiva: Factor común $\overline{A}\overline{B}$ en los últimos dos términos.
5	$F = BD + \overline{A}\overline{B}(\overline{C} + D)$	Ley Conmutativa: Reordenamiento dentro del paréntesis.
6	$F = BD + \overline{A}\overline{B}\overline{C}D$	Ley de De Morgan: $\overline{C} + D = \overline{C\overline{D}}$

Resultado final simplificado:

$$F = BD + \overline{A}\overline{B}(\overline{C} + D)$$

En este ejemplo se muestra la simplificación paso a paso de una expresión booleana de 4 términos, aplicando sistemáticamente las leyes distributiva, de complemento, de identidad, conmutativa y de De Morgan para llegar a una forma más compacta con 2 términos principales. Luego generamos el mapa de Karnaugh del ejercicio.

Mapa de Karnaugh con agrupamientos ($F = BD + \overline{A}\overline{B}(\overline{C} + D)$)

	00	01	11	10
00	0	0	0	0
01	0	1	1	1
11	0	1	1	1
10	1	1	1	1

La simplificación $F = BD + \overline{A}\overline{B}(\overline{C} + D)$, tiene **agrupamientos** en **azul**, BD (columnas $CD=01$ y $CD=11$ con $B=1$, filas $AB=01, 11$) y en **verde**, $\overline{A}\overline{B}$ (fila $AB=10$, que cubre todos los valores de CD excepto $CD=00$)

A continuación, se presenta otro ejemplo de simplificación de una expresión booleana con cuatro variables, utilizando las leyes y teoremas fundamentales, siendo la expresión original: $F = \overline{A}BC + A\overline{B}C + ABC + AB\overline{C}$, entonces su desarrollo y justificación sería el siguiente (ver Tabla 1.15):

Ejemplo de simplificación de expresiones booleanas (Método algebraico)

Tabla 1.15: Simplificación de $F = \overline{A}BC + A\overline{B}C + ABC + AB\overline{C}$

Paso	Expresión Resultante	Ley o Teorema Aplicado
Inicial	$F = \overline{A}BC + A\overline{B}C + ABC + AB\overline{C}$	Expresión Original
1	$F = \overline{A}C(\overline{B} + B) + AB(C + \overline{C})$	Ley Distributiva: Factor común $\overline{A}C$ en los primeros dos términos y AB en los últimos dos.
2	$F = \overline{A}C(1) + AB(1)$	Ley de Complemento (OR): $\overline{B} + B = 1$ y $C + \overline{C} = 1$
3	$F = \overline{A}C + AB$	Ley de Identidad (AND): $\overline{A}C \cdot 1 = \overline{A}C$ y $AB \cdot 1 = AB$

Resultado final simplificado:

$$F = \overline{A}C + AB$$

En este ejemplo se muestra la simplificación paso a paso de una expresión booleana de 4 términos, aplicando sistemáticamente las leyes distributiva, de complemento e identidad para llegar a una forma más compacta con solo 2 términos. Luego generamos el mapa de Karnaugh del ejercicio.

Mapa de Karnaugh con agrupamientos ($F = \overline{A}C + AB$)

	0	1
00	0	0
01	0	1
11	1	1
10	0	1

La simplificación $F = \overline{A}C + AB$, tiene **agrupamientos** en azul, AB (fila $AB=11$, que cubre ambos valores de C) y en verde, $\overline{A}C$ (columna $C=1$, filas $AB=01, 10$ donde $A=0$)

A continuación, se presenta otro ejemplo de simplificación de una expresión booleana con cuatro variables, utilizando las leyes y teoremas fundamentales, siendo la expresión original: $(F = \overline{A}\overline{B}C + ABC + \overline{A}B\overline{C} + \overline{A}BC)$, entonces su desarrollo y justificación sería el siguiente (ver Tabla 1.16):

Ejemplo de simplificación de expresiones booleanas (4 variables, método algebraico)

Tabla 1.16: Simplificación de $F = \overline{A}\overline{B}C + ABC + \overline{A}B\overline{C} + \overline{A}BC$

Paso	Expresión Resultante	Ley o Teorema Aplicado
Inicial	$F = \overline{A}\overline{B}C + ABC + \overline{A}B\overline{C} + \overline{A}BC$	Expresión Original (ordenada para agrupar términos semejantes)
1	$F = AC(\overline{B} + B) + \overline{A}B(\overline{C} + C)$	Ley Distributiva: factor común AC en los dos primeros términos y factor común $\overline{A}B$ en los dos últimos.
2	$F = AC(1) + \overline{A}B(1)$	Ley de Complemento (OR): $\overline{B} + B = 1$ y $\overline{C} + C = 1$.
3	$F = AC + \overline{A}B$	Ley de Identidad (AND): $AC \cdot 1 = AC$ y $\overline{A}B \cdot 1 = \overline{A}B$.

Resultado final simplificado:

$$F = AC + \overline{A}B$$

Además se presenta la tabla de verdad resultante (ver Tabla 1.17):

Tabla de verdad de $F = AC + \overline{A}B$

Tabla 1.17: Tabla de verdad de $F(A, B, C, D)$ con $F = AC + \overline{A}B$

A	B	C	D	F
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	0
1	1	1	0	1
1	1	1	1	1

Luego se procede a realizar el proceso de creación del mapa de Karnaugh del ejercicio.

Mapa de Karnaugh (4 variables) para F

AB	CD			
	00	01	11	10
00	0	0	0	0
01	1	1	1	1
11	0	0	1	1
10	0	0	1	1

En el mapa se identifican dos grupos mínimos:

- Toda la fila $AB = 01$ (donde $A = 0$ y $B = 1$), que corresponde al término $\overline{A}B$.
- Un grupo de cuatro celdas en las columnas $CD = 11$ y $CD = 10$ con $A = 1$ y $C = 1$, que corresponde a AC .

Por tanto la forma mínima es:

$$F = AC + \overline{A}B.$$

Nota sobre el consenso: El término de consenso entre AC y $\overline{A}B$ es BC . Sin embargo, este no reduce la expresión porque es redundante:

$$AC + \overline{A}B + BC = AC + \overline{A}B.$$

Por tanto la forma obtenida ya es mínima.

Luego se procede a generar el mapa de Karnaugh del ejercicio.

Mapa de Karnaugh con agrupamientos ($F = AC + \overline{A}B$)

	00	01	11	10
00	0	0	0	0
01	1	1	1	1
11	0	0	1	1
10	0	0	1	1

Resultado de la simplificación:

$$F = AC + \overline{A}B$$

A continuación, se presenta otro ejemplo de simplificación de una expresión booleana con cuatro variables, utilizando las leyes y teoremas fundamentales, siendo la expresión original: $(F = \overline{A}BCD + \overline{A}B\overline{C}D + AB\overline{C}D + ABC\overline{D})$, entonces su desarrollo y justificación sería el siguiente (ver Tabla 1.18):

Ejemplo de simplificación de expresiones booleanas (4 variables, método algebraico)

Tabla 1.18: Simplificación de $F = \overline{A}BCD + \overline{A}B\overline{C}D + AB\overline{C}D + ABC\overline{D}$

Paso	Expresión Resultante	Ley o Teorema Aplicado
Inicial	$F = \overline{A}BCD + \overline{A}B\overline{C}D + AB\overline{C}D + ABC\overline{D}$	Expresión Original (ordenada para agrupar términos semejantes)
1	$F = \overline{C}D(\overline{A}B + \overline{A}B + AB) + ABC\overline{D}$	Ley Distributiva: factor común $\overline{C}D$ en los tres primeros términos.
2	$F = \overline{C}D(\overline{A}(\overline{B} + B) + AB) + ABC\overline{D}$	Ley Distributiva: factor común $\overline{A}$ en los dos primeros términos del paréntesis.
3	$F = \overline{C}D(\overline{A}(1) + AB) + ABC\overline{D}$	Ley de Complemento (OR): $\overline{B} + B = 1$.
4	$F = \overline{C}D(\overline{A} + AB) + ABC\overline{D}$	Ley de Identidad (AND): $\overline{A} \cdot 1 = \overline{A}$.
5	$F = \overline{C}D(\overline{A} + B) + ABC\overline{D}$	Teorema de Absorción: $X + XY = X + Y$. Aquí, $\overline{A} + AB = \overline{A} + B$.
6	$F = \overline{C}D\overline{A}B + ABC\overline{D}$	Ley de De Morgan: $\overline{A} + B = \overline{\overline{A}B}$, equivalente a $\overline{A}B$ en forma simplificada.

Resultado final simplificado:

$$F = \overline{C}D(\overline{A} + B) + ABC\overline{D}$$

Además se presenta la tabla de verdad resultante (ver Tabla 1.19):

Tabla de verdad de $F = \overline{C}D(\overline{A} + B) + ABC\overline{D}$

Tabla 1.19: Tabla de verdad de $F(A, B, C, D)$ con $F = \overline{C}D(\overline{A} + B) + ABC\overline{D}$

A	B	C	D	F
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

Luego se procede a realizar el proceso de creación del mapa de Karnaugh del ejercicio.

Mapa de Karnaugh (4 variables) para F

AB	CD			
	00	01	11	10
00	0	1	0	0
01	0	1	0	0
11	0	1	0	1
10	0	0	0	0

En el mapa se identifican dos grupos mínimos:

- Un grupo vertical de tres celdas en la columna $CD = 01$ con filas $AB = 00, 01, 11$ (donde $C = 0$ y $D = 1$), que corresponde al término $\overline{C}D(\overline{A} + B)$.
- Una celda individual en $AB = 11, CD = 10$ (donde $A = 1, B = 1, C = 1, D = 0$), que corresponde a $AB\overline{C}\overline{D}$.

Por tanto la forma mínima es:

$$F = \overline{C}D(\overline{A} + B) + AB\overline{C}\overline{D}.$$

Nota: El término $\overline{C}D(\overline{A} + B)$ equivale a tres minitérminos: $\overline{A}\overline{C}D + B\overline{C}D + \overline{C}D$. La expresión ya se encuentra en su forma mínima.

Luego se procede a generar el mapa de Karnaugh del ejercicio.

Mapa de Karnaugh con agrupamientos ($F = \overline{C}D(\overline{A} + B) + AB\overline{C}\overline{D}$)

	00	01	11	10
00	0	1	0	0
01	0	1	0	0
11	0	1	0	1
10	0	0	0	0

Resultado de la simplificación:

$$F = \overline{C}D(\overline{A} + B) + AB\overline{C}\overline{D}$$

Ejemplos de proposiciones y no proposiciones

Tabla 1.20: Distinción entre proposiciones (con valor de verdad) y expresiones no proposicionales

¿Proposición?	Expresión o Enunciado	Valor / Razón
PROPOSICIONES (Se puede asignar V o F)		
Sí	“Quito es capital de Ecuador”	Verdadero (Hecho geográfico)
Sí	“Saturno es plano”	Falso (Hecho científico)
Sí	“El número 26 es impar”	Falso (Definición matemática)
Sí	$6 \times 5 > 15$	Falso (10 no es mayor que 15)
Sí	“En el año 2050, el mundo será perfecto”	Falso (Aseveración indemostrable y subjetiva)
NO PROPOSICIONES (No se puede asignar V o F)		
No	“¿Quién inventó el teléfono?”	No aplica (Pregunta o Interrogación)
No	“¡Cierra la puerta!”	No aplica (Imperativo u Orden)
No	$x + 2 = 5$	No aplica (Proposición Abierta, depende de x)
No	“Ojalá llueva mañana”	No aplica (Deseo o Exclamación)
No	“El color rojo corre más rápido”	No aplica (Juicio de valor/Ausencia de sentido lógico)

A continuación, presentamos unos ejemplos del tipo de conectivos lógicos fundamentales (binarios y unitarios) y sus tablas de verdad, definiendo sus símbolos formales para su lectura y utilización (ver Tabla 1.21).

Conectivos lógicos fundamentales y tablas de verdad

Tabla 1.21: Conectivos lógicos, resumen

Conectivo	Símbolo	Lectura	Definición	Tabla de Verdad
Operadores lógicos				
Negación	$\neg p, \sim p$	“no p ”	$\neg p$ es verdadera cuando p es falsa , y viceversa.	$V \rightarrow F, F \rightarrow V$
Conjunción	$p \wedge q$	“ p y q ”	$p \wedge q$ es verdadera solo cuando ambas proposiciones son verdaderas.	$VV \rightarrow V, VF \rightarrow F, FV \rightarrow F, FF \rightarrow F$
Disyunción (Inclusiva)	$p \vee q$	“ p o q ”	$p \vee q$ es verdadera cuando al menos una proposición es verdadera.	$VV \rightarrow V, VF \rightarrow V, FV \rightarrow V, FF \rightarrow F$
Implicación	$p \rightarrow q$	“si p , entonces q ”	$p \rightarrow q$ es falsa solo cuando p es verdadera (hipótesis) y q es falsa (conclusión).	$VV \rightarrow V, VF \rightarrow F, FV \rightarrow V, FF \rightarrow V$
Bicondicional	$p \leftrightarrow q$	“ p si y solo si q ”	$p \leftrightarrow q$ es verdadera cuando p y q tienen el mismo valor de verdad.	$VV \rightarrow V, VF \rightarrow F, FV \rightarrow F, FF \rightarrow V$
Disyunción Exclusiva	$p \oplus q$	“ p o q , pero no ambos”	$p \oplus q$ es verdadera cuando exactamente una de las proposiciones es verdadera.	$VV \rightarrow F, VF \rightarrow V, FV \rightarrow V, FF \rightarrow F$

1.2.1. Introducción

La teoría de proposiciones lógicas, forma parte de una rama fundamental de las matemáticas conocida como la lógica formal. Esta se centra en el estudio del argumento razonado, teniendo la unidad fundamental en las proposiciones y las relaciones que se establecen entre ellas, abstraídas de su contenido semántico particular. Esta teoría es fundamentada a partir de los trabajos de filósofos y matemáticos de la antigüedad. Su formalización moderna y su desarrollo se dio a partir del siglo XIX y principios del XX, con figuras como Gottlob Frege y Bertrand Russell, que permitieron colocar las bases para la lógica matemática y, por extensión, la ciencia de la computación, ver Fig. 1.15.

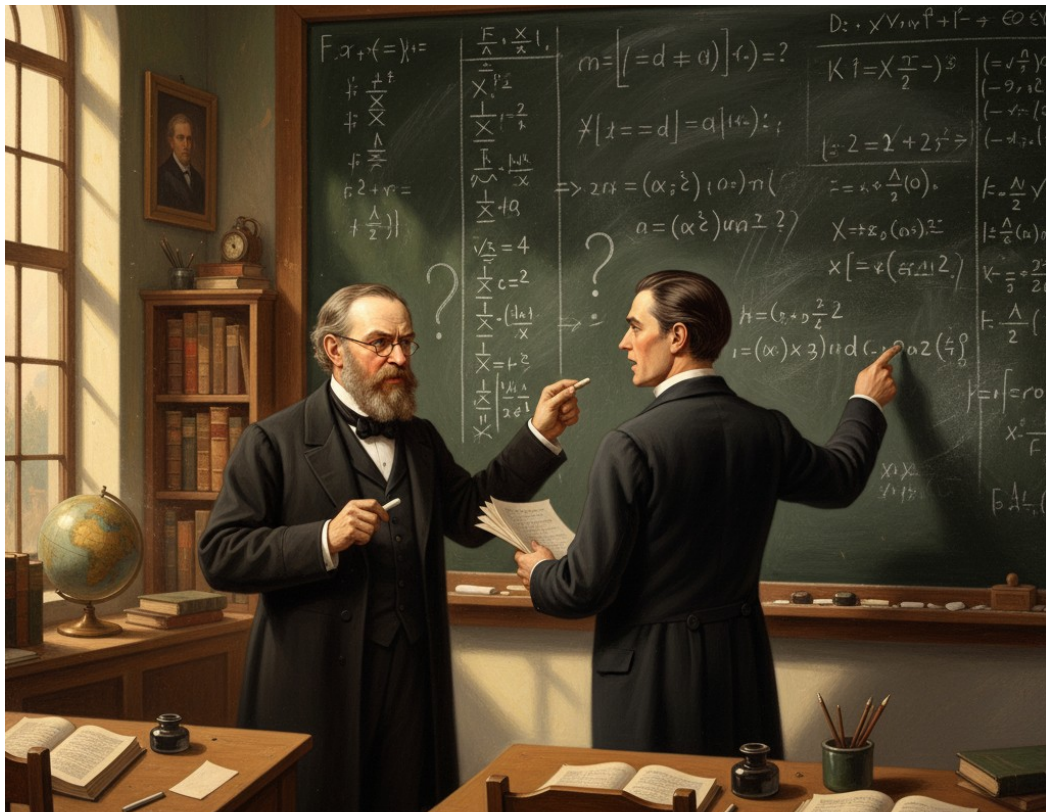


Figura 1.15: Imagen creada por la inteligencia artificial Gemini para la sección introducción, en la imagen se muestra imaginariamente a Gottlob Frege y Bertrand Russell.

En varias investigaciones actuales, en el ámbito del razonamiento computacional [Chajda & Laenger, 2025; Hu et al., 2025], han demostrado que, aunque la lógica proposicional clásica tiene limitaciones en su capacidad expresiva, la teoría proposicional sigue siendo el punto de partida para sistemas lógicos más complejos, incluyendo la lógica de predicados de primer orden. Así, la capacidad de esta teoría para modelar el razonamiento deductivo, tiene vigencia en la actualidad, y ha trascendido su origen filosófico para convertirse en una herramienta analítica indispensable en múltiples disciplinas [Llerena Izquierdo, 2023b].

El impacto de la lógica proposicional se manifiesta de manera profunda en las ciencias de la computación, donde su álgebra booleana constituye el cimiento de la arquitectura de los equipos y dispositivos (hardware), así como el diseño de circuitos digitales y los

lenguajes de programación (software) [Cao & Stepnicka, 2024; Llerena Izquierdo, 2020]. Los conectores lógicos ($\neg$, $\wedge$, $\vee$) corresponden directamente a las puertas lógicas (NOT, AND, OR), que son los componentes básicos de todos los sistemas electrónicos. Así, por ejemplo en su aplicación práctica, se evidencia en la verificación formal de sistemas, donde se utiliza para probar la corrección de diseños de hardware, garantizando que un circuito se comporte como se espera bajo todas las condiciones posibles [Fahmy-Hooke, 2024], ver Fig. 1.16.

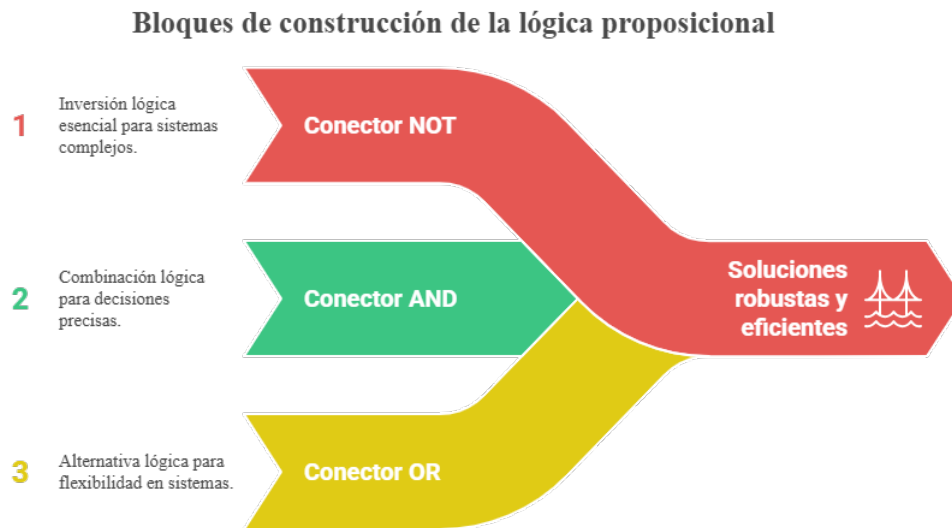


Figura 1.16: Imagen creada por la inteligencia artificial Napkin.ai para la sección introducción, en la imagen se muestra los bloques de construcción de la lógica.

Investigaciones de alto impacto, en la actualidad, destacan el uso de la lógica proposicional en el desarrollo de la inteligencia artificial, especialmente en los sistemas expertos y en la resolución de problemas lógicos dentro de los juegos, demostrando cómo su formalismo permite codificar el conocimiento humano para que sea procesado por máquinas [O’Sullivan, 2024]. El alcance actual de la lógica proposicional en la industria y la programación es amplio y a menudo de forma implícita. En la programación, las sentencias condicionales (if-then, if-else) y los bucles (while, for) se construyen directamente sobre los principios de la lógica proposicional [Faerna, 2024].

Los operadores lógicos en cualquier lenguaje de programación moderna son la base para la implementación directa de los conectivos proposicionales, permitiendo controlar el flujo de ejecución de los programas y tomar decisiones basadas en valores de verdad [Yue et al., 2024]. La utilidad del control de flujo, se extiende al diseño y optimización de algoritmos, a la creación de consultas de bases de datos que filtran información mediante expresiones lógicas y al desarrollo de sistemas de inteligencia artificial simbólica.

Además, la lógica proposicional ha trascendido a diversos ámbitos de estudio, donde su marco formal es utilizado para modelar y analizar razonamientos. En matemáticas, por ejemplo, es esencial para la construcción de demostraciones formales y la fundamentación de otras áreas como la teoría de conjuntos. En el campo de la inteligencia artificial, su aplicación es fundamental en el desarrollo de funciones de inferencia que, partiendo de un conjunto de hechos, patrones y reglas lógicas, generen nuevo conocimiento de manera algorítmica. Asimismo, en la ingeniería industrial, se utiliza para la optimización de

procesos, estableciendo relaciones lógicas entre variables para, por ejemplo, maximizar ganancias o minimizar costos en las líneas de producción, ver Fig. 1.17.

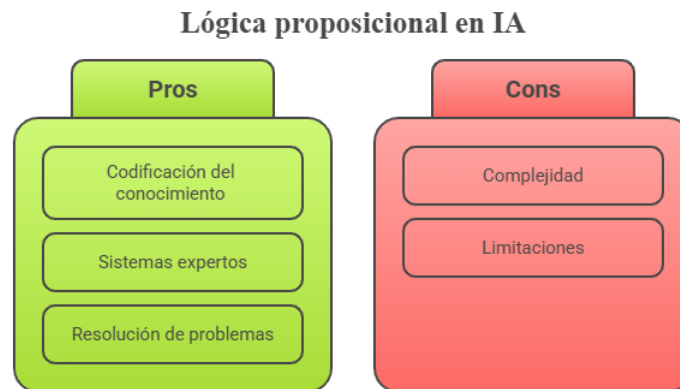


Figura 1.17: Imagen creada por la inteligencia artificial Napkin.ai para la sección introducción, en la imagen se muestra la lógica proposicional en IA.

La teoría de proposiciones lógicas ha evolucionado de un concepto filosófico a una herramienta computacional de análisis fundamental y de indiscutible relevancia en la era digital. Su impacto se evidencia en la estructura para la computación moderna, desde los circuitos más básicos hasta los sistemas de inteligencia artificial más avanzados [Cohen et al., 2024], ver Fig. 1.18.



Figura 1.18: Imagen creada por la inteligencia artificial Napkin.ai para la sección introducción, en la imagen se muestra la evolución de lógica proposicional.

Su alcance actual abarca la programación, la ingeniería y el desarrollo de software, donde sus principios rigen la toma de decisiones algorítmicas y la optimización de procesos. La trascendencia de esta teoría es tal que su formalismo lógico sigue siendo un pilar

indispensable para la innovación tecnológica y el desarrollo de nuevas áreas de investigación en la inteligencia artificial, la verificación de sistemas y la resolución de problemas complejos en una amplia gama de disciplinas.

1.2.2. Conectores proposicionales

En los últimos años, la industria de las tecnologías de información (TI) comparten soluciones y herramientas para verificar especificaciones formales de sistemas hardware y software. Se han dado cuenta del impacto y la importancia que tiene la investigación [Llerena-Izquierdo & Ayala-Carabayo, 2025b] para mejorar procesos de diseño e implementación que optimizan la planificación y la producción, ver Fig. 1.19.



Figura 1.19: Imagen del Centro de conocimiento e innovación deportiva del FC Barcelona que se enfoca a generar, compartir y aplicar conocimiento a través de proyectos de investigación.

Se requiere de formación en los conectivos lógicos para que los estudiantes tengan competencias en el uso, razonamiento y potenciación de sistemas (ver Tabla 1.22).

Conectivos lógicos fundamentales y tablas de verdad				
Tabla 1.22: Conectivos lógicos proposicionales				
Conectivo	Símbolo	Lectura	Definición	Tabla de Verdad
Operadores lógicos				
Negación	$\neg p$	“no p ”	$\neg p$ es verdadera cuando p es falsa , y viceversa.	$V \rightarrow F, F \rightarrow V$
Conjunción o AND	$p \wedge q$	“ p y q ”	$p \wedge q$ es verdadera solo cuando ambas proposiciones son verdaderas.	$VV \rightarrow V, VF \rightarrow F, FV \rightarrow F, FF \rightarrow F$
Disyunción u OR	$p \vee q$	“ p o q ” (inclusivo)	$p \vee q$ es verdadera cuando al menos una proposición es verdadera.	$VV \rightarrow V, VF \rightarrow V, FV \rightarrow V, FF \rightarrow F$
Implicación	$p \rightarrow q$	“si p , entonces q ”	$p \rightarrow q$ es falsa solo cuando p es verdadera (hipótesis) y q es falsa (conclusión).	$VV \rightarrow V, VF \rightarrow F, FV \rightarrow V, FF \rightarrow V$
Bicondicional o XNOR	$p \leftrightarrow q$	“ p si y solo si q ”	$p \leftrightarrow q$ es verdadera cuando p y q tienen el mismo valor de verdad.	$VV \rightarrow V, VF \rightarrow F, FV \rightarrow F, FF \rightarrow V$
Disyunción Exclusiva o XOR	$p \oplus q$	“ p o q , pero no ambos”	$p \oplus q$ es verdadera cuando exactamente una de las proposiciones es verdadera.	$VV \rightarrow F, VF \rightarrow V, FV \rightarrow V, FF \rightarrow F$

1.2.3. Las tablas de verdad

Las tablas de verdad permiten determinar los posibles resultados de combinaciones de verdadero o falso de una estructura proposicional propuesta. Sea λ un lenguaje que contiene dos proposiciones: p y q . Si consideramos el principio de bivalencia, p se puede representar mediante la siguiente imagen (ver Fig. 1.20).

Conectivo Lógico NOT (Formato Binario)		
Tabla de verdad NOT ($\neg p$)		
p	Operación	$\neg p$
1	$\neg 1$	0
0	$\neg 0$	1

Figura 1.20: Conectivo lógico binario fundamental, puerta lógica NOT.

Además de las diferentes tablas de conectivos lógicos en formato binario (ver Fig. 1.21).

Conectivos Lógicos (Formato Binario)			
Tabla de verdad AND ($p \wedge q$)			
p	q	Operación	$p \wedge q$
1	1	$1 \wedge 1$	1
1	0	$1 \wedge 0$	0
0	1	$0 \wedge 1$	0
0	0	$0 \wedge 0$	0
Tabla de verdad NAND ($\neg(p \wedge q)$)			
p	q	Operación	$p \text{ NAND } q$
1	1	$\neg(1 \wedge 1)$	0
1	0	$\neg(1 \wedge 0)$	1
0	1	$\neg(0 \wedge 1)$	1
0	0	$\neg(0 \wedge 0)$	1
Tabla de verdad OR ($p \vee q$)			
p	q	Operación	$p \vee q$
1	1	$1 \vee 1$	1
1	0	$1 \vee 0$	1
0	1	$0 \vee 1$	1
0	0	$0 \vee 0$	0
Tabla de verdad NOR ($\neg(p \vee q)$)			
p	q	Operación	$p \text{ NOR } q$
1	1	$\neg(1 \vee 1)$	0
1	0	$\neg(1 \vee 0)$	0
0	1	$\neg(0 \vee 1)$	0
0	0	$\neg(0 \vee 0)$	1
Tabla de verdad Implicación ($p \rightarrow q$)			
p	q	Operación	$p \rightarrow q$
1	1	$1 \rightarrow 1$	1
1	0	$1 \rightarrow 0$	0
0	1	$0 \rightarrow 1$	1
0	0	$0 \rightarrow 0$	1
Tabla de verdad XOR ($p \oplus q$)			
p	q	Operación	$p \oplus q$
1	1	$1 \oplus 1$	0
1	0	$1 \oplus 0$	1
0	1	$0 \oplus 1$	1
0	0	$0 \oplus 0$	0
Tabla de verdad Bicondicional ($p \leftrightarrow q$)			
p	q	Operación	$p \leftrightarrow q$
1	1	$1 \leftrightarrow 1$	1
1	0	$1 \leftrightarrow 0$	0
0	1	$0 \leftrightarrow 1$	0
0	0	$0 \leftrightarrow 0$	1
Tabla de verdad XNOR ($p \leftrightarrow q$ ó $\overline{p \oplus q}$)			
p	q	Operación	$p \text{ XNOR } q$
1	1	$1 \leftrightarrow 1$	1
1	0	$1 \leftrightarrow 0$	0
0	1	$0 \leftrightarrow 1$	0
0	0	$0 \leftrightarrow 0$	1

Figura 1.21: Conectivos lógicos binarios fundamentales, incluyendo las puertas lógicas (NAND, NOR, XOR, XNOR).

Las tablas de verdad de conectivos lógicos en formato lógico de verdadero y falso se representan como se muestra a continuación (ver Fig. 1.22).

Conectivos Lógicos (Formato Lógico)			
Tabla de verdad Negación ($\neg p$)			
p	Operación	$\neg p$	
V	$\neg V$	F	
F	$\neg F$	V	

Figura 1.22: Conectivo lógico fundamental (Verdadero/Falso), de la compuerta lógica NOT.

La tablas de verdad muestran en forma sistemática los valores de verdad de una proposición compuesta en función de todas las combinaciones posibles de los valores de verdad de las proposiciones que la componen (ver Fig. 1.23).

Conectivos Lógicos (Formato Lógico)			
Tabla de verdad Conjunción ($p \wedge q$)			
p	q	Operación	$p \wedge q$
V	V	$V \wedge V$	V
V	F	$V \wedge F$	F
F	V	$F \wedge V$	F
F	F	$F \wedge F$	F
Tabla de verdad NAND ($\neg(p \wedge q)$)			
p	q	Operación	$p \text{ NAND } q$
V	V	$\neg(V \wedge V)$	F
V	F	$\neg(V \wedge F)$	V
F	V	$\neg(F \wedge V)$	V
F	F	$\neg(F \wedge F)$	V
Tabla de verdad Disyunción ($p \vee q$)			
p	q	Operación	$p \vee q$
V	V	$V \vee V$	V
V	F	$V \vee F$	V
F	V	$F \vee V$	V
F	F	$F \vee F$	F
Tabla de verdad NOR ($\neg(p \vee q)$)			
p	q	Operación	$p \text{ NOR } q$
V	V	$\neg(V \vee V)$	F
V	F	$\neg(V \vee F)$	F
F	V	$\neg(F \vee V)$	F
F	F	$\neg(F \vee F)$	V
Tabla de verdad Implicación ($p \rightarrow q$)			
p	q	Operación	$p \rightarrow q$
V	V	$V \rightarrow V$	V
V	F	$V \rightarrow F$	F
F	V	$F \rightarrow V$	V
F	F	$F \rightarrow F$	V
Tabla de verdad XOR ($p \oplus q$)			
p	q	Operación	$p \oplus q$
V	V	$V \oplus V$	F
V	F	$V \oplus F$	V
F	V	$F \oplus V$	V
F	F	$F \oplus F$	F
Tabla de verdad Bicondicional ($p \leftrightarrow q$)			
p	q	Operación	$p \leftrightarrow q$
V	V	$V \leftrightarrow V$	V
V	F	$V \leftrightarrow F$	F
F	V	$F \leftrightarrow V$	F
F	F	$F \leftrightarrow F$	V
Tabla de verdad XNOR ($\overline{p \oplus q}$)			
p	q	Operación	$p \text{ XNOR } q$
V	V	$V \leftrightarrow V$	V
V	F	$V \leftrightarrow F$	F
F	V	$F \leftrightarrow V$	F
F	F	$F \leftrightarrow F$	V

Figura 1.23: Conectivos lógicos fundamentales (Verdadero/Falso), incluyendo las compuertas lógicas (NAND, NOR, XOR, XNOR).

Si analizamos las proposiciones p y q en conjunto, podremos observar que la tabla permite expresar todos los posibles valores de verdad de las mismas.

1.2.4. Equivalencias lógicas

El análisis de proposiciones requiere de formular las expresiones mediante un proceso de simplificación de los argumentos para obtener una conclusión (ver Tabla 1.23).

Leyes, Axiomas y Teoremas de la Teoría de Proposiciones			
Tabla 1.23: Leyes y equivalencias lógicas fundamentales			
Nombre de la Ley	Expresión Lógica	Ejemplo de Aplicación	Tipo
I. Leyes Fundamentales (Axiomas)			
Idempotencia	$p \wedge p \equiv p$ $p \vee p \equiv p$	$(p \vee r) \wedge (p \vee r) \equiv p \vee r$	Axioma
Doble Negación	$\neg(\neg p) \equiv p$	$\neg(\neg(q \vee r)) \equiv q \vee r$	Axioma
Conmutativa	$p \wedge q \equiv q \wedge p$ $p \vee q \equiv q \vee p$	$(p \vee r) \wedge q \equiv q \wedge (p \vee r)$	Axioma
Asociativa	$p \wedge (q \wedge r) \equiv (p \wedge q) \wedge r$	$p \vee (q \vee r) \equiv (p \vee q) \vee r$	Axioma
Distributiva	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$	Axioma
II. Leyes de Identidad y Complemento			
Identidad	$p \wedge V \equiv p$ $p \vee F \equiv p$	$(p \wedge q) \wedge V \equiv p \wedge q$	Teorema
Dominación	$p \vee V \equiv V$ $p \wedge F \equiv F$	$(p \vee q) \vee V \equiv V$	Teorema
Complemento	$p \vee \neg p \equiv V$ (Tercero Excluido)	$\neg(p \wedge q) \vee (p \wedge q) \equiv V$	Teorema
	$p \wedge \neg p \equiv F$ (No Contradicción)	$\neg(p \wedge q) \wedge (p \wedge q) \equiv F$	Teorema
III. Teoremas de Simplificación (Equivalencias)			
Absorción 1	$p \vee (p \wedge q) \equiv p$	$\neg q \vee (\neg q \wedge r) \equiv \neg q$	Teorema
Absorción 2	$p \wedge (p \vee q) \equiv p$	$p \wedge (\neg p \vee \neg r) \equiv p \wedge \neg r$	Teorema
De Morgan (Conj.)	$\neg(p \wedge q) \equiv \neg p \vee \neg q$	$\neg((\neg p) \wedge r) \equiv p \vee \neg r$	Equivalencia
De Morgan (Disy.)	$\neg(p \vee q) \equiv \neg p \wedge \neg q$	$\neg(\neg p \vee r) \equiv p \wedge \neg r$	Equivalencia
IV. Equivalencias de Conectivos (Artificios Lógicos)			
Implicación	$p \rightarrow q \equiv \neg p \vee q$	$(p \wedge q) \rightarrow r \equiv \neg(p \wedge q) \vee r$	Equivalencia
Bicondicional	$p \leftrightarrow q \equiv (p \rightarrow q) \wedge (q \rightarrow p)$	$p \leftrightarrow q \equiv (\neg p \vee q) \wedge (\neg q \vee p)$	Equivalencia
Contraposición	$p \rightarrow q \equiv \neg q \rightarrow \neg p$	$\neg p \rightarrow q \equiv \neg q \rightarrow p$	Equivalencia
V. Teoremas del Consenso (Resolución)			
Consenso	$(p \vee q) \wedge (\neg p \vee r) \wedge (q \vee r) \equiv (p \vee q) \wedge (\neg p \vee r)$	Si se tiene $A \vee B$ y $\neg A \vee C$, la cláusula $B \vee C$ es el consenso. $(p \vee q) \wedge (\neg p \vee r) \wedge (q \vee r)$ se reduce a $(p \vee q) \wedge (\neg p \vee r)$	Teorema
VI. Equivalencias de la Disyunción Exclusiva ($\oplus$)			
Disyunción Excl.	$p \oplus q \equiv (p \wedge \neg q) \vee (\neg p \wedge q)$	Se usa para eliminar el conectivo $\oplus$ y simplificar con $\wedge/\vee$.	Equivalencia
Negación de $\oplus$	$\neg(p \oplus q) \equiv p \leftrightarrow q$	La negación de la disyunción exclusiva es el bicondicional.	Equivalencia
VII. Reglas de Inferencia (Formas de Razonamiento)			
Modus Ponens	$[(p \rightarrow q) \wedge p] \rightarrow q$	$[(p \rightarrow q) \wedge p]$ puede ser reemplazado por q en la derivación.	Inferencia
Modus Tollens	$[(p \rightarrow q) \wedge \neg q] \rightarrow \neg p$	Si la conclusión es falsa, la hipótesis original debe ser falsa.	Inferencia
Silogismo Hipotético	$[(p \rightarrow q) \wedge (q \rightarrow r)] \rightarrow (p \rightarrow r)$	Si " $A \rightarrow B$ " y " $B \rightarrow C$ ", entonces " $A \rightarrow C$ ".	Inferencia
Simplificación	$(p \wedge q) \rightarrow p$	Si es verdad que $(p \wedge q)$, se puede inferir p como verdad.	Inferencia

1.2.5. Tipos de proposiciones compuestas

Las fórmulas contingentes, las contradicciones y las tautologías son conceptos muy importantes del Cálculo Proposicional Clásico (CPC). Una *tautología* (*Ley Lógica*) es una proposición compuesta que es siempre verdadera (V), sin importar los valores de verdad de las proposiciones simples que la componen. La columna final contiene solo valores Verdaderos (V) o 1. Representa una Ley Lógica cuya verdad es incuestionable en el sistema. Un ejemplo de ella es, la Ley del Tercero Excluido $p \vee \neg p$. Una *Contradicción* (*Falacia*) es una proposición compuesta que es siempre falsa (F), sin importar los valores de verdad de las proposiciones simples que la componen. La columna final contiene solo valores Falsos (F) o 0. Representa una imposibilidad o una inconsistencia lógica. Un ejemplo es, la Ley de no contradicción negada $p \wedge \neg p$. Una *Contingencia* (*Fórmula Neutra*) es una proposición compuesta cuyo valor de verdad depende de los valores de verdad de las proposiciones simples. No es ni una tautología ni una contradicción. La columna final contiene una mezcla de valores Verdaderos (V) y Falsos (F). Su verdad no está garantizada lógicamente; depende de las valoraciones de las fórmulas atómicas o proposiciones simples. Un ejemplo de ella es, Una simple implicación $p \rightarrow q$. Será V si p es V y q es V ($V \rightarrow V$). Será F si p es V y q es F ($V \rightarrow F$) (ver Tabla 1.24).

Clasificación Fundamental de Proposiciones Lógicas			
Tabla 1.24: Comportamiento en la columna final de la tabla de verdad			
Criterio	Tautología (✓)	Contradicción (×)	Contingencia (△)
Valor Final	Siempre Verdadera (V)	Siempre Falsa (F)	Depende (V y F)
Definición	Ley Lógica: Verdad garantizada.	Falacia Lógica: Falsedad garantizada.	Fórmula Neutra: Su verdad es circunstancial.
Composición T.V.	La columna final contiene solo valores V o 1.	La columna final contiene solo valores F o 0.	La columna final contiene ambos valores V y F.
Ejemplo	$p \vee \neg p$	$p \wedge \neg p$	$p \rightarrow q$

Entonces el análisis y la clasificación debe empezar primero por las proposiciones atómicas para establecer el valor final de la proposición compuesta, ver Fig. 1.24.

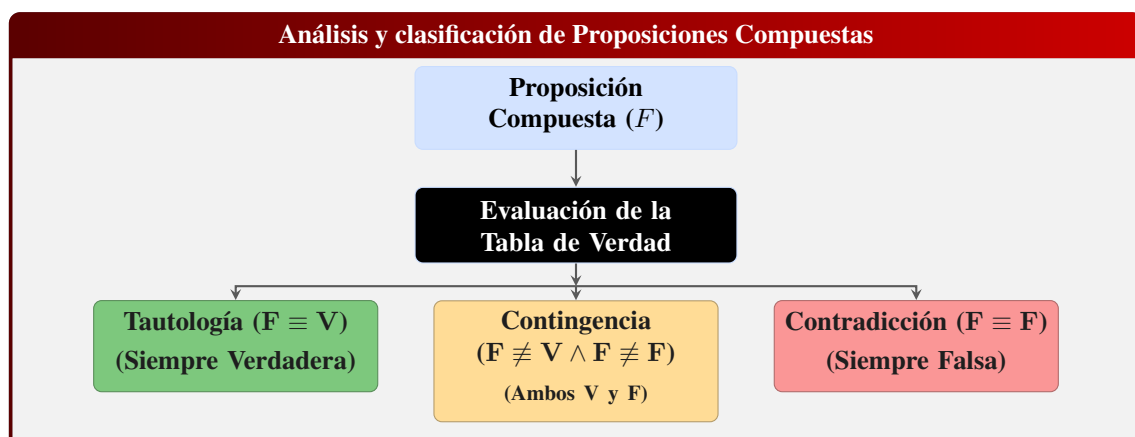


Figura 1.24: Diagrama de flujo que ilustra la clasificación de una proposición compuesta según el resultado de su tabla de verdad.

A continuación te presento un ejercicio donde se evidencia la evaluación de la proposición compuesta y su respuesta que es una **Contingencia** (ver Tabla 1.25).

Tabla de Verdad: Análisis de la Contingencia						
Tabla 1.25: Tabla de verdad de $(p \wedge q) \vee (q \wedge \neg r)$						
p	q	r	$p \wedge q$	$\neg r$	$q \wedge \neg r$	$(p \wedge q) \vee (q \wedge \neg r)$
V	V	V	V	F	F	V
V	V	F	V	V	V	V
V	F	V	F	F	F	F
V	F	F	F	V	F	F
F	V	V	F	F	F	F
F	V	F	F	V	V	V
F	F	V	F	F	F	F
F	F	F	F	V	F	F

A continuación te presento otro ejercicio donde se evidencia la evaluación de la proposición compuesta y su respuesta que es una **Contingencia** (ver Tabla 1.26).

Tabla de Verdad: Análisis de la Contingencia					
Tabla 1.26: Tabla de verdad de $(\neg p \wedge q) \rightarrow r$					
p	q	r	$\neg p$	$\neg p \wedge q$	$(\neg p \wedge q) \rightarrow r$
V	V	V	F	F	V
V	V	F	F	F	V
V	F	V	F	F	V
V	F	F	F	F	V
F	V	V	V	V	V
F	V	F	V	V	F
F	F	V	V	F	V
F	F	F	V	F	V

A continuación te presento otro ejercicio donde se evidencia la evaluación de la proposición compuesta y su respuesta que es una **Contingencia** (ver Tabla 1.27).

Tabla de Verdad: Análisis de la Contingencia				
Tabla 1.27: Tabla de verdad de $(p \leftrightarrow q) \wedge \neg p$				
p	q	$p \leftrightarrow q$	$\neg p$	$(p \leftrightarrow q) \wedge \neg p$
V	V	V	F	F
V	F	F	F	F
F	V	F	V	F
F	F	V	V	V

A continuación te presento otro ejercicio donde se evidencia la evaluación de la proposición compuesta y su respuesta que es una **Contingencia** (ver Tabla 1.28).

Tabla de Verdad: Análisis de la Contingencia

Tabla 1.28: Tabla de verdad de $(\neg p \vee q) \wedge \neg q$

p	q	$\neg p$	$\neg q$	$\neg p \vee q$	$(\neg p \vee q) \wedge \neg q$
V	V	F	F	V	F
V	F	F	V	F	F
F	V	V	F	V	F
F	F	V	V	V	V

A continuación te presento un ejercicio con cuatro variables proposicionales donde se evidencia la evaluación de la proposición compuesta y su respuesta que es una **Contingencia** (ver Tabla 1.29).

Tabla de Verdad: Contingencia con Cuatro Variables

Tabla 1.29: Tabla de verdad de $((p \wedge q) \rightarrow r) \leftrightarrow \neg s$

p	q	r	s	$p \wedge q$	$(p \wedge q) \rightarrow r$	$\neg s$	$((p \wedge q) \rightarrow r) \leftrightarrow \neg s$
V	V	V	V	V	V	F	F
V	V	V	F	V	V	V	V
V	V	F	V	V	F	F	V
V	V	F	F	V	F	V	F
V	F	V	V	F	V	F	F
V	F	V	F	F	V	V	V
V	F	F	V	F	V	F	F
V	F	F	F	F	V	V	V
F	V	V	V	F	V	F	F
F	V	V	F	F	V	V	V
F	V	F	V	F	V	F	F
F	V	F	F	F	V	V	V
F	F	V	V	F	V	F	F
F	F	V	F	F	V	V	V
F	F	F	V	F	V	F	F
F	F	F	F	F	V	V	V

A continuación te presento un ejercicio donde se evidencia la evaluación de la proposición compuesta y su respuesta que es una **Contingencia** (ver Tabla 1.30).

Tabla de Verdad: Análisis de la Contingencia

Tabla 1.30: Tabla de verdad de $p \rightarrow (q \wedge \neg r)$

p	q	r	$\neg r$	$q \wedge \neg r$	$p \rightarrow (q \wedge \neg r)$
V	V	V	F	F	F
V	V	F	V	V	V
V	F	V	F	F	F
V	F	F	V	F	F
F	V	V	F	F	V
F	V	F	V	V	V
F	F	V	F	F	V
F	F	F	V	F	V

A continuación te presento un ejercicio donde se evidencia la evaluación de la proposición compuesta y su respuesta que es una **Tautología** (ver Tabla 1.31).

Tabla de Verdad: Análisis de la Tautología					
Tabla 1.31: Tabla de verdad de $(\neg p \wedge q) \rightarrow (p \rightarrow q)$					
p	q	$\neg p$	$\neg p \wedge q$	$p \rightarrow q$	$(\neg p \wedge q) \rightarrow (p \rightarrow q)$
V	V	F	F	V	V
V	F	F	F	F	V
F	V	V	V	V	V
F	F	V	F	V	V

A continuación te presento otro ejercicio donde se evidencia la evaluación de la proposición compuesta y su respuesta que es una **Tautología** (ver Tabla 1.32).

Tabla de Verdad: Análisis de la Tautología (Modus Tollens)						
Tabla 1.32: Tabla de verdad de $[(p \rightarrow q) \wedge \neg q] \rightarrow \neg p$						
p	q	$\neg p$	$\neg q$	$p \rightarrow q$	$(p \rightarrow q) \wedge \neg q$	$[(p \rightarrow q) \wedge \neg q] \rightarrow \neg p$
V	V	F	F	V	F	V
V	F	F	V	F	F	V
F	V	V	F	V	F	V
F	F	V	V	V	V	V

A continuación te presento un ejercicio con tres variables que evidencia la evaluación de la proposición compuesta y su respuesta, que es una **Tautología** (ver Tabla 1.33).

Tabla de Verdad: Análisis de la Tautología (Principio de Resolución)							
Tabla 1.33: Tabla de verdad de $[(\neg p \vee q) \wedge (p \vee r)] \rightarrow (q \vee r)$							
p	q	r	$\neg p$	$\neg p \vee q$	$p \vee r$	$(\neg p \vee q) \wedge (p \vee r)$	$[(\neg p \vee q) \wedge (p \vee r)] \rightarrow (q \vee r)$
V	V	V	F	V	V	V	V
V	V	F	F	V	V	V	V
V	F	V	F	F	V	F	V
V	F	F	F	F	V	F	V
F	V	V	V	V	V	V	V
F	V	F	V	V	F	F	V
F	F	V	V	V	V	V	V
F	F	F	V	V	F	F	V

A continuación te presento un ejercicio con dos variables que evidencia la evaluación de la proposición compuesta y su respuesta, que es una **Tautología** (ver Tabla 1.34).

Tabla de Verdad: Análisis de la Tautología (De Morgan Condicional)							
Tabla 1.34: Tabla de verdad de $\neg(p \vee q) \rightarrow (\neg p \wedge \neg q)$							
p	q	$p \vee q$	$\neg(p \vee q)$	$\neg p$	$\neg q$	$\neg p \wedge \neg q$	$\neg(p \vee q) \rightarrow (\neg p \wedge \neg q)$
V	V	V	F	F	F	F	V
V	F	V	F	F	V	F	V
F	V	V	F	V	F	F	V
F	F	F	V	V	V	V	V

A continuación te presento un ejercicio con dos variables que evidencia la evaluación de la proposición compuesta y su respuesta, que es una **Contradicción** (ver Tabla 1.35).

Tabla de Verdad: Análisis de la Contradicción

Tabla 1.35: Tabla de verdad de $p \wedge \neg p$

p	$\neg p$	$p \wedge \neg p$
V	F	F
F	V	F

A continuación te presento otro ejercicio con dos variables que evidencia la evaluación de la proposición compuesta y su respuesta, que es una **Contradicción** (ver Tabla 1.36).

Tabla de Verdad: Análisis de la Contradicción

Tabla 1.36: Tabla de verdad de $(p \leftrightarrow q) \wedge (p \oplus q)$

p	q	$p \leftrightarrow q$	$p \oplus q$	$(p \leftrightarrow q) \wedge (p \oplus q)$
V	V	V	F	F
V	F	F	V	F
F	V	F	V	F
F	F	V	F	F

A continuación te presento un ejercicio con tres variables que evidencia la evaluación de la proposición compuesta y su respuesta, que es una **Contradicción** (ver Tabla 1.37).

Tabla de Verdad: Análisis de la Contradicción

Tabla 1.37: Tabla de verdad de $(p \wedge q) \wedge \neg(p \wedge r)$

p	q	r	$p \wedge q$	$p \wedge r$	$\neg(p \wedge r)$	$(p \wedge q) \wedge \neg(p \wedge r)$
V	V	V	V	V	F	F
V	V	F	V	F	V	F
V	F	V	F	V	F	F
V	F	F	F	F	V	F
F	V	V	F	F	V	F
F	V	F	F	F	V	F
F	F	V	F	F	V	F
F	F	F	F	F	V	F

A continuación te presento un ejercicio con dos variables que evidencia la evaluación de la proposición compuesta y su respuesta, que es una **Contradicción** (ver Tabla 1.38).

Tabla de Verdad: Análisis de la Contradicción

Tabla 1.38: Tabla de verdad de $(p \rightarrow q) \wedge (p \wedge \neg q)$

p	q	$p \rightarrow q$	$\neg q$	$p \wedge \neg q$	$(p \rightarrow q) \wedge (p \wedge \neg q)$
V	V	V	F	F	F
V	F	F	V	V	F
F	V	V	F	F	F
F	F	V	V	F	F

A continuación te presento un ejercicio con cuatro variables que evidencia la evaluación de la proposición compuesta y su respuesta, que es una **Contradicción** (ver Tabla 1.39).

Tabla de Verdad: Contradicción con Cuatro Variables

Tabla 1.39: Tabla de verdad de $(p \wedge q \wedge r \wedge s) \wedge \neg(p \wedge q \wedge r \wedge s)$

p	q	r	s	$P = (p \wedge q \wedge r \wedge s)$	$\neg P$	$P \wedge \neg P$
V	V	V	V	V	F	F
V	V	V	F	F	V	F
V	V	F	V	F	V	F
V	V	F	F	F	V	F
V	F	V	V	F	V	F
V	F	V	F	F	V	F
V	F	F	V	F	V	F
V	F	F	F	F	V	F
F	V	V	V	F	V	F
F	V	V	F	F	V	F
F	V	F	V	F	V	F
F	V	F	F	F	V	F
F	F	V	V	F	V	F
F	F	V	F	F	V	F
F	F	F	V	F	V	F
F	F	F	F	F	V	F

A continuación te presento un ejercicio con cinco variables que evidencia la evaluación de la proposición compuesta y su respuesta, que es una **Contradicción** (ver Tabla 1.40).

Tabla de Verdad: Contradicción con Cinco Variables

Tabla 1.40: Tabla de verdad de $(p \wedge q \wedge r \wedge s \wedge t) \wedge \neg(p \wedge q \wedge r \wedge s \wedge t)$

p	q	r	s	t	$P = (\dots \wedge t)$	$\neg P$	$P \wedge \neg P$
V	V	V	V	V	V	F	F
V	V	V	V	F	F	V	F
V	V	V	F	V	F	V	F
V	V	V	F	F	F	V	F
V	V	F	V	V	F	V	F
V	V	F	V	F	F	V	F
V	V	F	F	V	F	V	F
V	V	F	F	F	F	V	F
V	F	V	V	V	F	V	F
V	F	V	V	F	F	V	F
V	F	V	F	V	F	V	F
V	F	V	F	F	F	V	F
V	F	F	V	V	F	V	F
V	F	F	V	F	F	V	F
V	F	F	F	V	F	V	F
V	F	F	F	F	F	V	F
F	V	V	V	V	F	V	F
F	V	V	V	F	F	V	F
F	V	V	F	V	F	V	F
F	V	V	F	F	F	V	F
F	V	F	V	V	F	V	F
F	V	F	V	F	F	V	F
F	V	F	F	V	F	V	F
F	V	F	F	F	F	V	F
F	F	V	V	V	F	V	F
F	F	V	V	F	F	V	F
F	F	V	F	V	F	V	F
F	F	V	F	F	F	V	F
F	F	F	V	V	F	V	F
F	F	F	V	F	F	V	F
F	F	F	F	V	F	V	F
F	F	F	F	F	F	V	F

A continuación te presento un ejercicio donde la proposición compuesta se extrae de una situación cotidiana, y su evaluación final resulta ser una **Contingencia** (ver Tabla 1.41).

1. Proposición Narrativa: La proposición compuesta se extrae de la siguiente situación cotidiana: “Si hoy es lunes y no he tenido una reunión, entonces me iré a casa temprano.”

2. Definición de Variables: Para la formalización, se definen las siguientes variables, p : Hoy es lunes; q : He tenido una reunión; r : Me iré a casa temprano.

3. Lenguaje Formal: Con estas definiciones, la proposición se traduce al lenguaje formal como $(p \wedge \neg q) \rightarrow r$.

Tabla de Verdad: Análisis de la Proposición Narrativa (Contingencia)

Tabla 1.41: Tabla de verdad de $(p \wedge \neg q) \rightarrow r$

p	q	r	$\neg q$	$p \wedge \neg q$	$(p \wedge \neg q) \rightarrow r$
V	V	V	F	F	V
V	V	F	F	F	V
V	F	V	V	V	V
V	F	F	V	V	F
F	V	V	F	F	V
F	V	F	F	F	V
F	F	V	V	F	V
F	F	F	V	F	V

A continuación te presento un ejercicio donde la proposición compuesta se extrae de una situación cotidiana y su evaluación final resulta ser una **Tautología** (ver Tabla 1.42).

1. Proposición Narrativa: La proposición compuesta se extrae de la siguiente situación cotidiana: “Si hay tráfico en la ciudad, entonces la universidad tendrá clases o la universidad no tendrá clases.”

2. Definición de Variables: Para la formalización, se definen las siguientes variables: p : La universidad tendrá clases; q : Hay tráfico en la ciudad.

3. Lenguaje Formal: Con estas definiciones, la proposición se traduce al lenguaje formal como $q \rightarrow (\neg p \vee p)$.

Tabla de Verdad: Análisis de la Proposición Narrativa (Tautología)

Tabla 1.42: Tabla de verdad de $q \rightarrow (\neg p \vee p)$

p	q	$\neg p$	$\neg p \vee p$	$q \rightarrow (\neg p \vee p)$
V	V	F	V	V
V	F	F	V	V
F	V	V	V	V
F	F	V	V	V

A continuación te presento un ejercicio donde la proposición compuesta se extrae de una situación cotidiana y su evaluación final resulta ser una **Contradicción** (ver Tabla 1.43).

- 1. Proposición Narrativa:** La proposición compuesta se extrae de la siguiente situación cotidiana: *"Hoy no estoy de vacaciones y estoy trabajando, y al mismo tiempo no estoy trabajando."*
- 2. Definición de Variables:** Para la formalización, se definen las siguientes variables: p : Estoy trabajando; q : Estoy de vacaciones.
- 3. Lenguaje Formal:** Con estas definiciones, la proposición se traduce al lenguaje formal como $(\neg q \wedge p) \wedge \neg p$.

Tabla de Verdad: Análisis de la Proposición Narrativa (Contradicción)

Tabla 1.43: Tabla de verdad de $(\neg q \wedge p) \wedge \neg p$

p	q	$\neg q$	$\neg p$	$\neg q \wedge p$	$(\neg q \wedge p) \wedge \neg p$
V	V	F	F	F	F
V	F	V	F	V	F
F	V	F	V	F	F
F	F	V	V	F	F

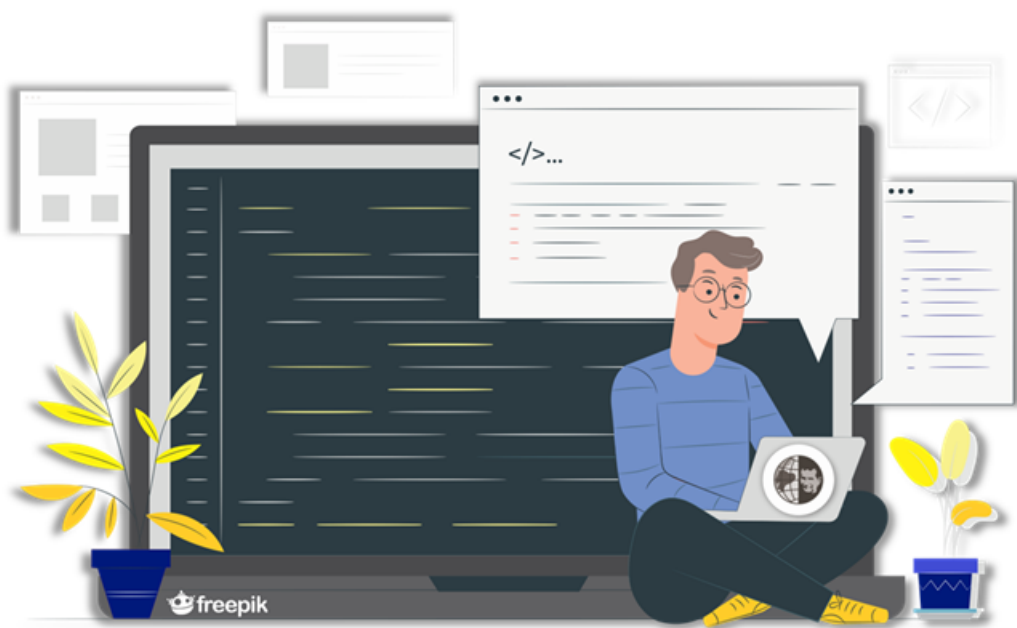
A continuación te presento un ejercicio donde la proposición compuesta se extrae de una situación cotidiana y su evaluación final resulta ser una **Contingencia** (ver Tabla 1.44).

- 1. Proposición Narrativa:** La proposición compuesta se extrae de la siguiente situación cotidiana: *"Si hace sol o tenemos tiempo libre, entonces iremos a la playa."*
- 2. Definición de Variables:** Para la formalización, se definen las siguientes variables: p : Hace sol; q : Tenemos tiempo libre; r : Iremos a la playa.
- 3. Lenguaje Formal:** Con estas definiciones, la proposición se traduce al lenguaje formal como $(p \vee q) \rightarrow r$.

Tabla de Verdad: Análisis de la Proposición Narrativa (Contingencia)

Tabla 1.44: Tabla de verdad de $(p \vee q) \rightarrow r$

p	q	r	$p \vee q$	$(p \vee q) \rightarrow r$
V	V	V	V	V
V	V	F	V	F
V	F	V	V	V
V	F	F	V	F
F	V	V	V	V
F	V	F	V	F
F	F	V	F	V
F	F	F	F	V



Laboratorio 1

Capítulo 1

1.3. Ejercicios del Laboratorio 1

A continuación empezamos con los ejercicios del laboratorio 1 de las prácticas de experimentación del capítulo 1.

1.3.1. Sección 1 del laboratorio 1

A continuación, responde las siguientes preguntas luego de observar y comparar las diferentes respuestas existentes para consolidar los conceptos fundamentales.

Para la **pregunta 1. ¿Qué es el Álgebra de Boole?**, te presento un conjunto de resúmenes que debes comparar y analizar para escribir tu respuesta, ver Tabla 1.45.

Análisis consolidado a la pregunta ¿Qué es el Álgebra de Boole?

Tabla 1.45: Comparativa de la utilidad del Álgebra Booleana en diferentes artefactos

Fuente	Respuestas	Explicación de utilidad y propósito
Gemini	Opera con 0 (Falso) y 1 (Verdadero). Utiliza las operaciones básicas AND ($\wedge$), OR ($\vee$) y NOT ($\neg$).	Es el <i>sistema matemático fundamental</i> de la lógica digital y la informática. Su importancia radica en que sus operaciones se implementan en <i>compuertas lógicas</i> y el <i>hardware</i> de la computadora.
Claude	Opera con 0 / 1 e incluye operaciones: AND ($\cdot$), OR ($+$), NOT ($\bar{A}$), además de Leyes de De Morgan y de Complemento, etc.	Es la <i>base fundamental</i> de la computación digital moderna. Se utiliza para <i>diseñar</i> y <i>optimizar circuitos</i> (procesadores, memorias) y escribir programas mediante expresiones lógicas.
Copilot	Trabaja con valores <i>Binarios</i> (0 y 1). Sus operaciones (AND, OR, NOT) se basan en la teoría de la electrónica digital.	Constituye la <i>base teórica</i> de la electrónica digital. Clave para la creación de <i>compuertas lógicas</i> , tablas de verdad y la <i>simplificación de circuitos</i> y lógica de sistemas.
Deepseek	Manipula variables con 1/0 (V/F) a partir de las operaciones lógicas fundamentales: AND, OR, NOT.	Rama matemática base del diseño de <i>circuitos digitales</i> , <i>arquitectura de computadoras</i> y <i>programación</i> , permitiendo modelar y simplificar la lógica del hardware y software.
ChatGPT	Trabaja con 0 y 1 (Falso o Verdadero, Apagado/Encendido), utilizando las operaciones lógicas básicas (AND, OR, NOT).	Su importancia radica en que constituye la <i>base del diseño de circuitos digitales</i> y de la <i>computación moderna</i> , permitiendo combinar y simplificar expresiones lógicas.

Ahora, responde con tus propias palabras y trata de recordar lo aprendido.

Para la **pregunta 2. ¿Qué es una compuerta lógica?**, te presento un conjunto de resúmenes que debes comparar y analizar para escribir tu respuesta, ver Tabla 1.46.

Análisis consolidado a la pregunta ¿Qué es una compuerta lógica?

Tabla 1.46: Resumen de las funciones y tipos de las compuertas lógicas

Fuente	Respuestas	Explicación de implementación
Gemini	Es el <i>bloque de construcción fundamental</i> de un circuito digital. Realiza una <i>operación lógica</i> elemental sobre entradas binarias (1 o 0) con una única salida, siguiendo el Álgebra de Boole.	Son implementadas físicamente mediante <i>transistores</i> . Se combinan para construir <i>funciones complejas</i> , permitiendo a los dispositivos electrónicos tomar decisiones y procesar información.
Claude	Dispositivo electrónico que implementa físicamente una operación booleana (AND, OR, NOT, NAND, NOR, XOR, XNOR) con una o más señales de entrada binarias.	Son los <i>bloques de construcción fundamentales</i> de todos los sistemas digitales modernos: procesadores, memorias, calculadoras. Se implementan usando <i>transistores</i> en circuitos integrados (chips).
Copilot	Dispositivo electrónico fundamental que realiza <i>operaciones lógicas</i> sobre una o más señales de entrada para producir una salida, basándose en reglas del Álgebra de Boole.	Se usan para construir <i>sistemas complejos</i> como procesadores y memorias, ya que permiten tomar <i>decisiones binarias</i> (0 o 1) según las combinaciones de entrada.
Deepseek	Circuito electrónico que opera con <i>señales binarias</i> (0 y 1) para implementar las <i>funciones básicas</i> del Álgebra de Boole (AND, OR, NOT).	Sirven como <i>bloques de construcción esenciales</i> para formar <i>circuitos digitales más complejos</i> (microprocesadores, memorias) y todos los sistemas digitales modernos.
ChatGPT	Dispositivo electrónico que realiza <i>operaciones lógicas básicas</i> (AND, OR y NOT) a partir de una o varias <i>señales de entrada binarias</i> (0 y 1) para producir una salida también binaria.	Son los <i>bloques fundamentales</i> en el diseño de <i>circuitos digitales</i> y permiten procesar información en computadoras, calculadoras y sistemas electrónicos en general.

Ahora, responde con tus propias palabras y trata de recordar lo aprendido.

Para la **pregunta 3. ¿Qué son los 0s y 1s lógicos y cuáles son los rangos de voltajes de cada uno?**, te presento un conjunto de resúmenes que debes comparar y analizar para escribir tu respuesta, ver Tabla 1.47.

Análisis consolidado: 0s y 1s lógicos y rangos de voltaje

Tabla 1.47: Correspondencia entre estados lógicos y rangos de voltaje en electrónica digital

Fuente	Definición del estado lógico	Rangos de voltaje y tecnologías
Gemini	<i>0 Lógico:</i> Estado bajo, falso o apagado. <i>1 Lógico:</i> Estado alto, verdadero o encendido. Ambos son la base de la señal digital en Álgebra de Boole.	<i>0 Lógico:</i> Voltaje cercano a 0 V. <i>1 Lógico:</i> Voltaje alto; típicamente 5 V (TTL) o 3,3 V (CMOS).
Claude	<i>0 Lógico:</i> “Falso”, “apagado”, “bajo”. <i>1 Lógico:</i> “Verdadero”, “encendido”, “alto”. Son representaciones binarias de estados discretos.	<i>Tecnología TTL (5V):</i> 0 L es 0 V a 0,8 V. 1 L es 2 V a 5 V. Menciona la <i>zona de incertidumbre</i> y tecnologías menores (3,3 V, 1,8 V).
Copilot	Representan los dos estados posibles (nivel bajo: 0 y nivel alto: 1) para procesar y almacenar información binaria en sistemas digitales.	<i>TTL:</i> 0 L es 0 V a 0,8 V. 1 L es 2 V a 5 V. <i>CMOS:</i> 0 L es $< 1/3$ de V_{CC} y 1 L es $> 2/3$ de V_{CC} .
Deepseek	Representaciones abstractas de dos estados discretos (0, falso o bajo y 1, verdadero o alto). Se materializan físicamente como rangos de voltaje.	<i>TTL:</i> 0 L es 0 a 0,8 V. 1 L es 2 a 5 V. <i>CMOS (3.3V):</i> 0 L es 0 a 0,9 V. Existe una <i>región prohibida</i> para garantizar claridad.
ChatGPT	<i>0 Lógico:</i> Estado bajo, falso o apagado. <i>1 Lógico:</i> Estado alto, verdadero o encendido. Ambos son los dos estados posibles de una señal digital en el álgebra de Boole.	<i>0 Lógico:</i> Voltaje cercano a 0 V. <i>1 Lógico:</i> Voltaje alto; típicamente 5 V (TTL) o 3,3 V (CMOS).

Ahora, responde con tus propias palabras y trata de recordar lo aprendido.

1.3.2. Sección 2 del laboratorio 1.

A continuación, investiga toda información que te permita comprender el funcionamiento de las siguientes compuertas.

Consolidación del aprendizaje

Compuertas Lógicas Digitales

[Investiga, observa, reflexiona, analiza y completa la información de las principales compuertas.]

Tabla: Compuerta NOT

Compuerta: NOT

Describe su funcionamiento:

Símbolo	Diagrama lógico	Tabla de verdad (2 entradas)	Circuito integrado	Código TTL

Tabla: Compuerta AND

Compuerta: AND

Describe su funcionamiento:

Símbolo	Diagrama lógico	Tabla de verdad (2 entradas)	Circuito integrado	Código TTL

Tabla: Compuerta OR

Compuerta: OR

Describe su funcionamiento:

Símbolo	Diagrama lógico	Tabla de verdad (2 entradas)	Circuito integrado	Código TTL

Tabla: Compuerta NAND

Compuerta: NAND

Describe su funcionamiento:

Símbolo	Diagrama lógico	Tabla de verdad (2 entradas)	Circuito integrado	Código TTL

Tabla: Compuerta NOR

Compuerta: NOR

Describe su funcionamiento:

Símbolo	Diagrama lógico	Tabla de verdad (2 entradas)	Circuito integrado	Código TTL

Tabla: Compuerta XOR

Compuerta: XOR

Describe su funcionamiento:

Símbolo	Diagrama lógico	Tabla de verdad (2 entradas)	Circuito integrado	Código TTL

Tabla: Compuerta XNOR

Compuerta: XNOR

Describe su funcionamiento:

Símbolo	Diagrama lógico	Tabla de verdad (2 entradas)	Circuito integrado	Código TTL

1.3.3. Sección 3 del laboratorio 1.

Ahora utilice un simulador de circuitos (puede usar Logisim u otro) y presente cinco ejemplos de circuitos combinatorios que usen compuertas lógicas e indicando qué hacen (ver Fig. 1.25).

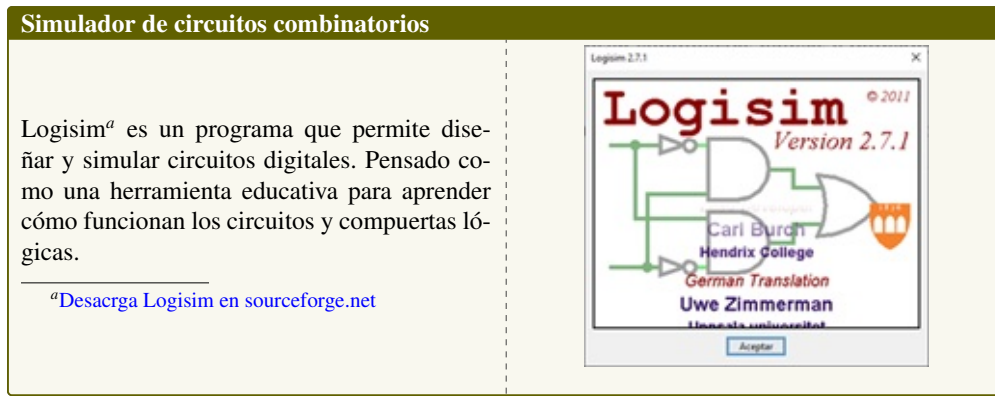


Figura 1.25: El programa de Logisim se descarga en sourceforge.net.

A continuación, reconocemos el ambiente de trabajo del programa y utilizamos las compuertas visibles, ver Fig. 1.26.

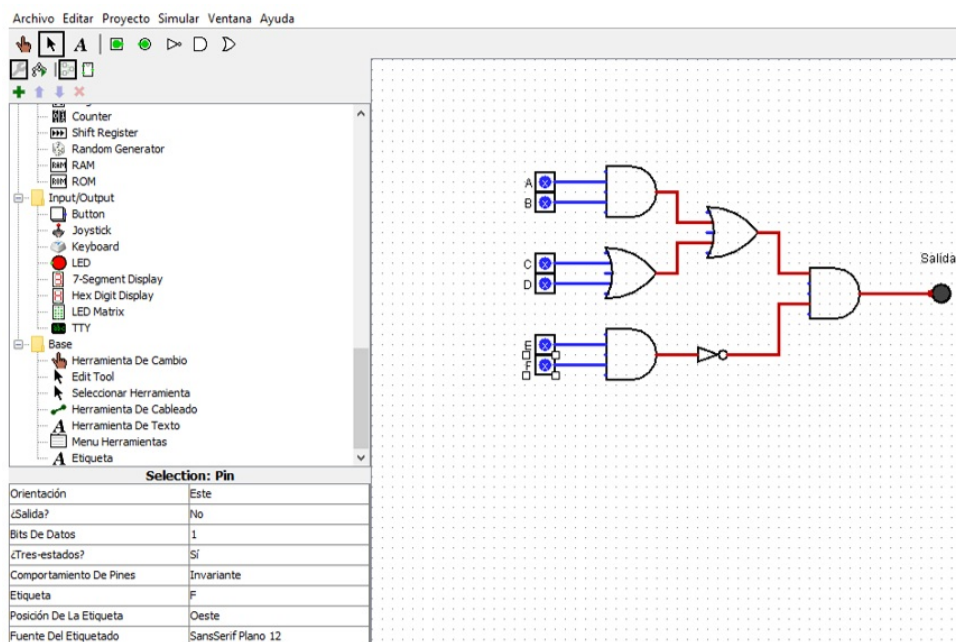


Figura 1.26: Entorno de trabajo de Logisim

En el menú del programa² puedes utilizar el módulo de “Ayuda” donde se encuentran elementos de asistencia para iniciar su uso. Finalmente, generamos varios esquemas con la utilización de las compuertas lógicas.

²[Descarga Logisim de su sitio web](#)

Se puede observar que la imagen visible en el entorno tiene diversos colores, a diferencia de la imagen cuando se exporta en formatos jpg, Gif o png, en el módulo de “Archivo”, exportar imagen, ver Fig. 1.27.

Para analizar y reproducir

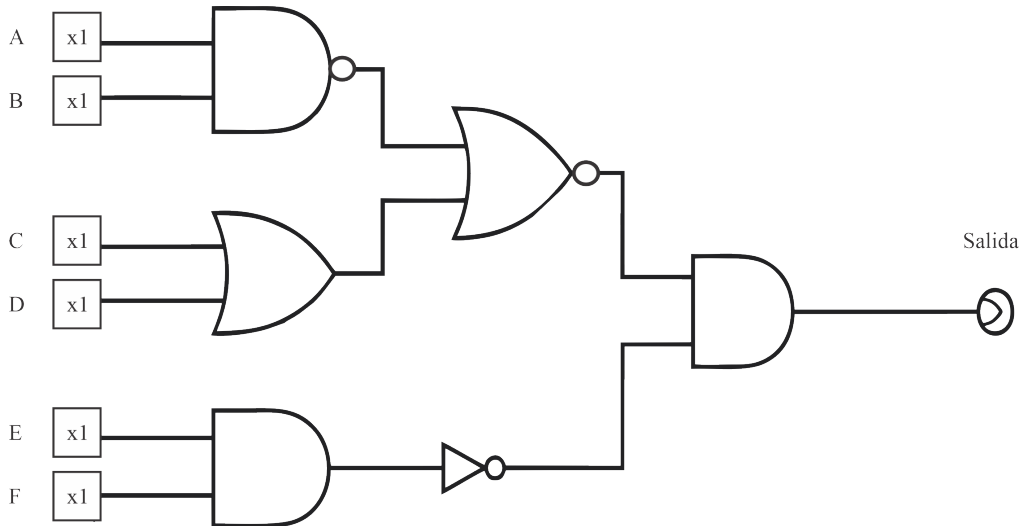


Figura 1.27: Imagen de salida del programa Logisim.

A continuación, en el espacio vacío de la siguiente caja dibuja tu diseño logrado luego del uso del programa Logisim.

Coloca tu dibujo en este espacio

A continuación, en el espacio vacío de la siguiente caja dibuja otro diseño logrado luego del uso del programa Logisim.

Coloca tu dibujo en este espacio



A continuación, en el espacio vacío de la siguiente caja dibuja un nuevo diseño logrado luego del uso del programa Logisim.

Coloca tu dibujo en este espacio



A continuación, en el espacio vacío de la siguiente caja dibuja otro diseño logrado luego del uso del programa Logisim.

Coloca tu dibujo en este espacio



A continuación, en el espacio vacío de la siguiente caja dibuja un nuevo diseño logrado luego del uso del programa Logisim.

Coloca tu dibujo en este espacio



1.3.4. Sección 4 del laboratorio 1.

Utilizando las propiedades y teoremas del Álgebra de Boole verifique si la siguiente igualdad es verdadera o falsa. Elabora las simplificaciones necesarias y enuncia cada una a medida que vayas desarrollando hacia abajo.

Para el ejercicio: $x'z + x'yz = x'z$, el proceso de desarrollo se encuentra a continuación, ver Tabla 1.48.

Verificación de igualdad booleana (Método Algebraico)

Tabla 1.48: Verificación de la igualdad $x'z + x'yz = x'z$

Paso	Expresión resultante	Ley o Teorema aplicado
L.I. Inicial	$x'z + x'yz$	Expresión Original (Lado Izquierdo)
1	$x'z(1 + y)$	Factor Común (Propiedad Distributiva): Se factoriza $x'z$.
2	$x'z(1)$	Ley de Dominación (o Unidad): $1 + A = 1$. Cualquier término sumado a 1 es 1.
3	$x'z$	Ley de Identidad: $A \cdot 1 = A$. Cualquier término multiplicado por 1 es el propio término.
L.D. Final	$x'z$	Igual al Lado Derecho (L.D.). La igualdad es VERDADERA.

Para el ejercicio de verificación de igualdad booleana: $x(x + y) = x$, el proceso de desarrollo se encuentra a continuación, ver Tabla 1.49.

Verificación de igualdad booleana (Método Algebraico)

Tabla 1.49: Verificación de la igualdad $x(x + y) = x$

Paso	Expresión resultante	Ley o Teorema aplicado
L.I. Inicial	$x(x + y)$	Expresión Original (Lado Izquierdo)
1	$x \cdot x + x \cdot y$	Ley Distributiva: Se multiplica x por cada término dentro del paréntesis.
2	$x + xy$	Ley de Idempotencia: $A \cdot A = A$. El término $x \cdot x$ se simplifica a x .
3	$x(1 + y)$	Factor Común (Propiedad Distributiva): Se factoriza x de ambos términos.
4	$x(1)$	Ley de Dominación (o Unidad): $1 + A = 1$. Cualquier cosa sumada a 1 es 1.
L.D. Final	x	Igual al Lado Derecho (L.D.). La igualdad es VERDADERA.

Utilizando las propiedades y teoremas del Álgebra de Boole, verifique si la siguiente igualdad es verdadera o falsa. Elabore las simplificaciones necesarias y enuncie cada ley o teorema aplicado a medida que desarrolle la expresión:

$$\overline{A}B\overline{C}D + \overline{A}B\overline{C}\overline{D} + A\overline{B}\overline{C}D + A\overline{B}C\overline{D} = \overline{C}D$$

Nota: Considere la notación $\overline{A}$ para la negación (NOT) y la simple yuxtaposición ($\overline{AB}$) para la conjunción (AND), ver Tabla 1.50.

Verificación de igualdad booleana (4 Variables)

Tabla 1.50: Verificación de la igualdad $\overline{A}B\overline{C}D + \overline{A}B\overline{C}\overline{D} + A\overline{B}\overline{C}D + A\overline{B}C\overline{D} = \overline{C}D$

Paso	Expresión resultante	Ley o Teorema aplicado
L.I. Inicial	$\overline{A}B\overline{C}D + \overline{A}B\overline{C}\overline{D} + A\overline{B}\overline{C}D + A\overline{B}C\overline{D}$	Expresión Original (Lado Izquierdo)
1	$\overline{C}D(\overline{A}B + \overline{A}B + A\overline{B} + A\overline{B})$	Factor Común (Propiedad Distributiva). Se factoriza $\overline{C}D$ de todos los términos.
2	$\overline{C}D(\overline{A}B + AB + A\overline{B})$	Ley de Idempotencia: $X + X = X$. El término $\overline{A}B$ se repite.
3	$\overline{C}D[B(\overline{A} + A) + A\overline{B}]$	Factor Común (Propiedad Distributiva). Se factoriza B de los dos primeros términos.
4	$\overline{C}D[B(1) + A\overline{B}]$	Ley de Complemento (OR): $\overline{A} + A = 1$.
5	$\overline{C}D(B + A\overline{B})$	Ley de Identidad: $X \cdot 1 = X$.
6	$\overline{C}D(B + A)$	Teorema de Absorción: $X + \overline{X}Y = X + Y$. Aquí, $B + A\overline{B} = B + A$.
7	$\overline{C}D$	Ley de Dominación (AND): $\overline{C}D \cdot (B + A)$ no simplifica a $\overline{C}D$, no es igual.
L.D. Final	$\overline{C}D$	Reagrupamos términos de la expresión original para optimizar u alternativa... (ver abajo)

A continuación, la simplificación del Lado Izquierdo ($\overline{C}D(\overline{A}B + AB + A\overline{B})$) buscando otra alternativa a la que se hizo en la Tabla 1.50, pero necesitamos verificar los últimos pasos, ver Tabla 1.51.

Continuación del análisis de la simplificación, otra alternativa

Tabla 1.51: Verificación detallada de la simplificación (continuación)

Paso	Expresión resultante	Ley o Teorema aplicado
Inicial (desde el paso 2)	$\overline{C}D \cdot (\overline{A}B + AB + A\overline{B})$	Continuación del Paso 2 de la Tabla 1.50
3	$\overline{C}D \cdot [B(\overline{A} + A) + A\overline{B}]$	Factor Común de B en los dos primeros términos.
4	$\overline{C}D \cdot [B(1) + A\overline{B}]$	Ley de Complemento (OR): $\overline{A} + A = 1$.
5	$\overline{C}D \cdot (B + A\overline{B})$	Ley de Identidad: $X \cdot 1 = X$.
6	$\overline{C}D \cdot (B + A)$	Teorema de Absorción: $X + \overline{X}Y = X + Y$. Aquí, $B + A\overline{B} = B + A$.
L.I. Simplificada	$\overline{C}D \cdot (B + A)$	Lado Izquierdo completamente simplificado.
L.D. Final	$\overline{C}D$	Se concluye que no son iguales.

Conclusión y corrección del enunciado: El Lado Izquierdo se simplifica a $\overline{C}D(A + B)$.

- La igualdad original $\overline{A}B\overline{C}D + \overline{A}B\overline{C}\overline{D} + A\overline{B}\overline{C}D + A\overline{B}C\overline{D} = \overline{C}D$ es **FALSA**.
- Para que la igualdad fuera **VERDADERA**, el Lado Derecho debería haber sido $\overline{C}D(A + B)$.

1.3.5. Sección 5 del laboratorio 1

Utilizando las propiedades y teoremas del Álgebra de Boole escriba el enunciado E en forma de suma de productos, $E = (((a'b')c)'((a' + c)(b' + c'))')'$

Aplice los siguientes pasos: (a). Use las *leyes De Morgan* y la *involución* para eliminar los complementos de los paréntesis hasta que se aplique solamente a variables. (b). Use la *ley distributiva* para transformar E en suma de productos. (c). Use las *leyes conmutativas*, de *idempotencia* y de *complemento* transformar cada producto de E en 0 o en un producto fundamental. (d). Use la *ley de absorción* para dejar E en la forma correcta de suma de productos.

Se presenta a continuación el desarrollo paso a paso, ver Tabla 1.52.

Ejercicio de simplificación a suma de productos

Tabla 1.52: Conversión de $E = (((a'b')c)'((a' + c)(b' + c'))')'$ a Suma de Productos

Paso	Expresión Resultante	Ley o Teorema Aplicado
Inicial	$E = (((a'b')c)'((a' + c)(b' + c'))')'$	Expresión Original
a. Uso de Leyes De Morgan e Involución		
a.1	$E = [(a'b')' + c'] + [(a' + c)' + (b' + c')']$	Ley De Morgan: $(XY)' = X' + Y'$ (aplicada a la conjunción principal).
a.2	$E = [(a'b')' + c'] + [(a''c') + (b''c'')]$	Ley De Morgan: $(X + Y)' = X'Y'$ (aplicada dentro del segundo corchete).
a.3	$E = [(a'' + b'') + c'] + [(ac') + (bc)]$	Ley De Morgan y Involución: $(a'b')' = a'' + b''$. También: $a'' = a$ y $c''' = c'$.
a.4	$E = [a + b + c'] + [ac' + bc]$	Ley de Involución: $a'' = a$ y $b'' = b$.
b. Uso de Ley Distributiva (para suma de productos)		
b.1	$E = a + b + c' + ac' + bc$	Eliminando corchetes, ya que todas las operaciones restantes son OR (suma).
c. Uso de Leyes Conmutativas, Idempotencia, y Complemento		
c.1	$E = a + b + c' + ac' + bc$	La expresión ya está en forma de suma de productos (Minterms y Maxterms).
d. Uso de Ley de Absorción		
d.1	$E = a + b + c'$	Ley de Absorción: $X + XY = X$. • $c' + ac' = c'$ (Término ac' es absorbido por c'). • $b + bc = b$ (Término bc es absorbido por b).
Final	$E = a + b + c'$	Expresión Final Simplificada en Suma de Productos.

A continuación, utilizando las propiedades y teoremas del Álgebra de Boole, simplifique la siguiente expresión lógica F a su forma mínima de Suma de Productos (SOP, sum of products), $F = (A + B)(\bar{A} + C)(B + C)$.

Aplice los siguientes pasos: (a). Use la *ley distributiva* para expandir los primeros dos paréntesis. (b). Simplifique la expresión resultante utilizando la *ley de complemento* ($X\bar{X} = 0$) y la *ley de identidad* ($X + 0 = X$). (c). Use la *ley de consenso* para simplificar la expresión por completo.

Se presenta a continuación el desarrollo paso a paso, ver Tabla 1.53.

Ejercicio de simplificación a suma de productos (Ley de Consenso)

Tabla 1.53: Simplificación de $F = (A + B)(\bar{A} + C)(B + C)$ a su mínima expresión SOP

Paso	Expresión Resultante	Ley o Teorema Aplicado
Inicial	$F = (A + B)(\bar{A} + C)(B + C)$	Expresión Original
a. Expansión de los primeros dos paréntesis		
a.1	$F = (A\bar{A} + AC + BA + BC)(B + C)$	Ley Distributiva: $X(Y + Z) = XY + XZ$.
b. Simplificación usando Complemento e Identidad		
b.1	$F = (0 + AC + \bar{A}B + BC)(B + C)$	Ley de Complemento (AND): $A\bar{A} = 0$.
b.2	$F = (AC + \bar{A}B + BC)(B + C)$	Ley de Identidad (OR): $X + 0 = X$.
c. Uso de la Ley de Consenso (Avanzada)		
c.1	$F = (AC + \bar{A}B)(B + C)$	Ley de Consenso (Forma OR): $XY + \bar{X}Z + YZ = XY + \bar{X}Z$. El término BC es redundante.
d. Conversión final a SOP		
d.1	$F = AC(B + C) + \bar{A}B(B + C)$	Ley Distributiva (Expandiendo la conjunción principal).
d.2	$F = ACB + ACC + \bar{A}BB + \bar{A}BC$	Ley Distributiva (Completando la expansión).
d.3	$F = ABC + AC + \bar{A}B + \bar{A}BC$	Ley de Idempotencia: $BB = B$ y $CC = C$.
d.4	$F = AC(B + 1) + \bar{A}B(1 + C)$	Factor Común (Agrupando términos semejantes).
d.5	$F = AC(1) + \bar{A}B(1)$	Ley de Dominación (Unidad): $1 + X = 1$.
Final	$F = AC + \bar{A}B$	Ley de Identidad: $X \cdot 1 = X$.

A continuación, utilizando las propiedades y teoremas del Álgebra de Boole, simplifique la siguiente expresión lógica G a su forma mínima de Suma de Productos (SOP), $G = AB + \bar{A}C + BC$.

Aplique los siguientes pasos: (a). Identifique el término redundante utilizando la *ley de consenso*. Demuestre su redundancia añadiendo el factor $1 = (A + \bar{A})$ al término. (b). Use la *ley distributiva* y las *leyes de complemento* para expandir y agrupar el término expandido con los términos originales. (c). Use la *ley de absorción* y la *ley de idempotencia* para eliminar los términos redundantes y obtener la expresión simplificada final.

Observe, analice y siga el proceso presentado en el diagrama de flujo, ver Fig. 1.28.

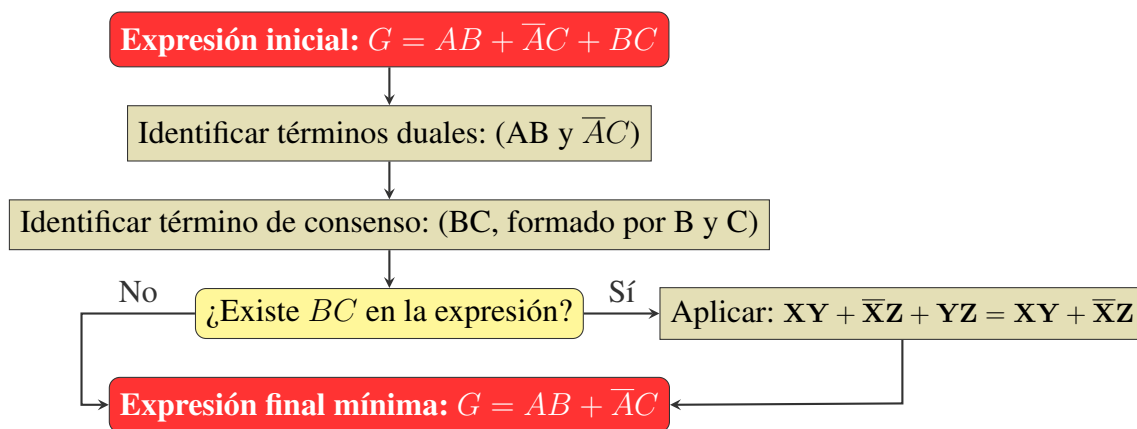


Figura 1.28: Proceso a seguir en el diagrama de flujo para la Ley de Consenso.

Se presenta a continuación el desarrollo del ejercicio paso a paso, ver Tabla 1.54.

Desarrollo: Simplificación por Ley de Consenso

Tabla 1.54: Simplificación de $G = AB + \overline{A}C + BC$ a su mínima expresión SOP

Paso	Expresión Resultante	Ley o Teorema Aplicado
Inicial	$G = AB + \overline{A}C + BC$	Expresión Original
a. Identificación y Demostración de Redundancia		
a.1	$G = AB + \overline{A}C + BC \cdot (A + \overline{A})$	Ley de Complemento: $A + \overline{A} = 1$. (Término BC es el consenso)
b. Expansión y Agrupación de Términos		
b.1	$G = AB + \overline{A}C + ABC + \overline{A}BC$	Ley Distributiva (Expandiendo el término BC).
b.2	$G = (AB + ABC) + (\overline{A}C + \overline{A}BC)$	Ley Asociativa (Reordenando para agrupar términos semejantes).
b.3	$G = AB(1 + C) + \overline{A}C(1 + B)$	Factor Común (Propiedad Distributiva).
b.4	$G = AB(1) + \overline{A}C(1)$	Ley de Dominación (Unidad): $1 + X = 1$.
c. Eliminación de Redundancia y Simplificación Final		
c.1	$G = AB + \overline{A}C$	Ley de Identidad: $X \cdot 1 = X$.
Final	$G = AB + \overline{A}C$	Ley de Consenso: $AB + \overline{A}C + BC = AB + \overline{A}C$. El término BC es redundante.

Finalmente, utilizando las propiedades y teoremas del Álgebra de Boole, escriba el enunciado H en su forma mínima de Suma de Productos (SOP), $H = (A + \overline{B}C)(A + \overline{B}C')$.

Aplique los siguientes pasos: (a). Use la *ley distributiva (inversa)* para factorizar el término común A . (b). Simplifique la expresión resultante utilizando la *ley distributiva (expansión)* y la *ley de complemento*. (c). Use la *ley de identidad* para obtener la expresión final en la forma mínima de Suma de Productos.

Se presenta a continuación el desarrollo paso a paso, ver Tabla 1.55.

Desarrollo: Simplificación por Distribución Inversa

Tabla 1.55: Simplificación de $H = (A + \overline{B}C)(A + \overline{B}C')$

Paso	Expresión Resultante	Ley o Teorema Aplicado
Inicial	$H = (A + \overline{B}C)(A + \overline{B}C')$	Expresión Original
a. Uso de Ley Distributiva (Inversa)		
a.1	$H = A + (\overline{B}C)(\overline{B}C')$	Ley Distributiva Inversa: $(X + Y)(X + Z) = X + YZ$.
b. Uso de Leyes de Complemento e Idempotencia		
b.1	$H = A + (\overline{B}\overline{B})(CC')$	Ley Asociativa/Conmutativa (Reordenando términos).
b.2	$H = A + (\overline{B})(0)$	Ley de Idempotencia: $\overline{B}\overline{B} = \overline{B}$. Ley de Complemento: $CC' = 0$.
b.3	$H = A + 0$	Ley de Dominación (Cero): $X \cdot 0 = 0$.
c. Uso de Ley de Identidad		
c.1	$H = A$	Ley de Identidad: $X + 0 = X$.
Final	$H = A$	Expresión Final Mínima en Suma de Productos.

1.3.6. Sección 6 del laboratorio 1

Utilizando el siguiente circuito que se muestra en la imagen a continuación, ver Fig. 1.29.

Para analizar y desarrollar el problema

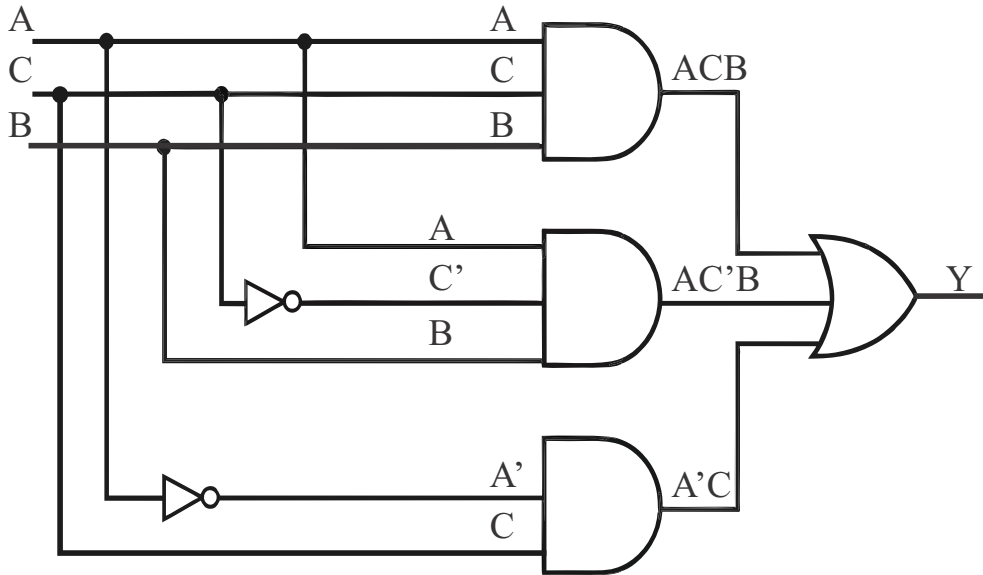


Figura 1.29: Imagen del programa Logisim.

Realice lo siguiente: a. Escriba la expresión booleana para el siguiente circuito. b. Construya la tabla de verdad. c. Simplifique la expresión resultante de la tabla de verdad utilizando los mapas de Karnaugh. d. Simplifique la expresión booleana de salida del circuito utilizando las leyes del álgebra de Boole. (Las dos expresiones simplificadas deben resultar iguales). e. Diseñe el circuito simplificado utilizando Logisim, guarde el archivo e imagen con la documentación de esta tarea.

Para el literal **a. Escriba la expresión booleana para el siguiente circuito**, la respuesta es la siguiente: $Y = ACB + AC'B + A'C$.

Realizando un análisis y tabla de verdad de $Y = ACB + AC'B + A'C$ se tendría la siguiente información, ver Tabla 1.56.

a. Simplificación por Álgebra de Boole

Tabla 1.56: Simplificación de la expresión Y

Paso	Expresión Resultante	Ley o Teorema Aplicado
Inicial	$Y = ACB + AC'B + A'C$	Expresión Original
1	$Y = AB(C + C') + A'C$	Ley Distributiva (Factor Común AB).
2	$Y = AB(1) + A'C$	Ley de Complemento: $C + C' = 1$.
Final	$Y = AB + A'C$	Ley de Identidad: $X \cdot 1 = X$.

Para el literal **b. Construya la tabla de verdad**, se tendría la siguiente información, ver Tabla 1.57.

b. Tabla de Verdad

Tabla 1.57: Tabla de verdad para la expresión simplificada $Y = AB + A'C$

Entradas			Salidas Intermedias y Final		
A	B	C	AB	A'C	$Y = AB + A'C$
0	0	0	0	0	0
0	0	1	0	1	1
0	1	0	0	0	0
0	1	1	0	1	1
1	0	0	0	0	0
1	0	1	0	0	0
1	1	0	1	0	1
1	1	1	1	0	1

Para el literal **c. Simplifique la expresión resultante de la tabla de verdad utilizando los mapas de Karnaugh**, se tendría la siguiente información, ver Tabla 1.58.

c. Simplificación por Mapa de Karnaugh (K-map)

Tabla 1.58: Mapa de Karnaugh y Simplificación de $Y = AB + \bar{A}C$

Paso	Descripción	Expresión Mínima
Inicial	Minterms de Y : $m(1, 3, 6, 7)$.	$Y = ACB + ACB + \bar{A}C$
1. Simplificación Algebraica Previa		
a.1	$Y = AB(C + C) + \bar{A}C$	$Y = AB + \bar{A}C$
2. Agrupación en el Mapa de Karnaugh (K-map)		
b.1	El mapa muestra los 1 en las celdas 1, 3, 6, 7.	
b.2	Grupo 1 (Rojo): Agrupa los 1 en $m(6, 7)$ (donde $A = 1$ y $B = 1$).	
b.3	Grupo 2 (Azul): Agrupa los 1 en $m(1, 3)$ (donde $A = 0$ y $C = 1$).	
3. Obtención de la expresión mínima		
Final	La suma de los dos grupos primos es la expresión mínima.	$Y_{\text{Mínima}} = AB + \bar{A}C$

El mapa de Karnaugh o K-map resultante es el siguiente:

Mapa de Karnaugh para $Y = AB + \bar{A}C$

	00 ($\bar{B}\bar{C}$)	01 ($\bar{B}C$)	11 (BC)	10 ($B\bar{C}$)
0 ($\bar{A}$)	0	1	1	0
1 (A)	0	0	1	1

Resultado de la simplificación por agrupación:

- **Grupo Azul (AB):** Celdas (110) y (111) → Simplifica a AB .
- **Grupo Rojo ($\bar{A}C$):** Celdas (001) y (011) → Simplifica a $\bar{A}C$.

$$Y_{\text{Mínima}} = AB + A'C$$

Para el literal **d. Simplifique la expresión booleana de salida del circuito utilizando las leyes del álgebra de Boole. (Las dos expresiones simplificadas deben resultar iguales)**, se tendría la siguiente información, ver Tabla 1.59.

Demostración de Equivalencia y Simplificación

Tabla 1.59: Simplificación de $Y = ACB + AC'B + A'C$

Paso	Expresión Resultante	Ley o Teorema Aplicado
Inicial	$Y = ACB + AC'B + A'C$	Expresión dada
1	$Y = (AB)C + (AB)C' + A'C$	Ley Conmutativa o Asociativa (Reagrupando términos).
2	$Y = AB(C + C') + A'C$	Ley Distributiva (Factor Común AB).
3	$Y = AB(1) + A'C$	Ley de Complemento: $C + C' = 1$.
Final	$Y = AB + A'C$	Ley de Identidad: $X \cdot 1 = X$.

Conclusión: La expresión simplificada es: $Y_{\text{Simplificada}} = AB + A'C$.

Esta expresión simplificada ya se había obtenido previamente por el Mapa de Karnaugh (K-map), confirmando la igualdad:

$$Y_{\text{Inicial}} \equiv Y_{\text{K-map}} \equiv Y_{\text{Simplificada}}$$

Para el literal **e. Diseñe el circuito simplificado utilizando Logisim, guarde el archivo e imagen con la documentación de esta tarea**, se tendría la siguiente información, ver Fig. 1.30.

Diseño final del circuito

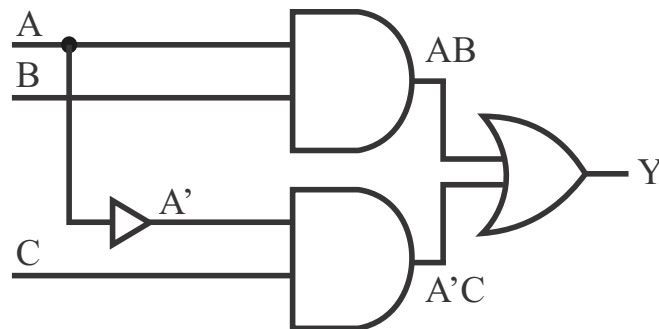
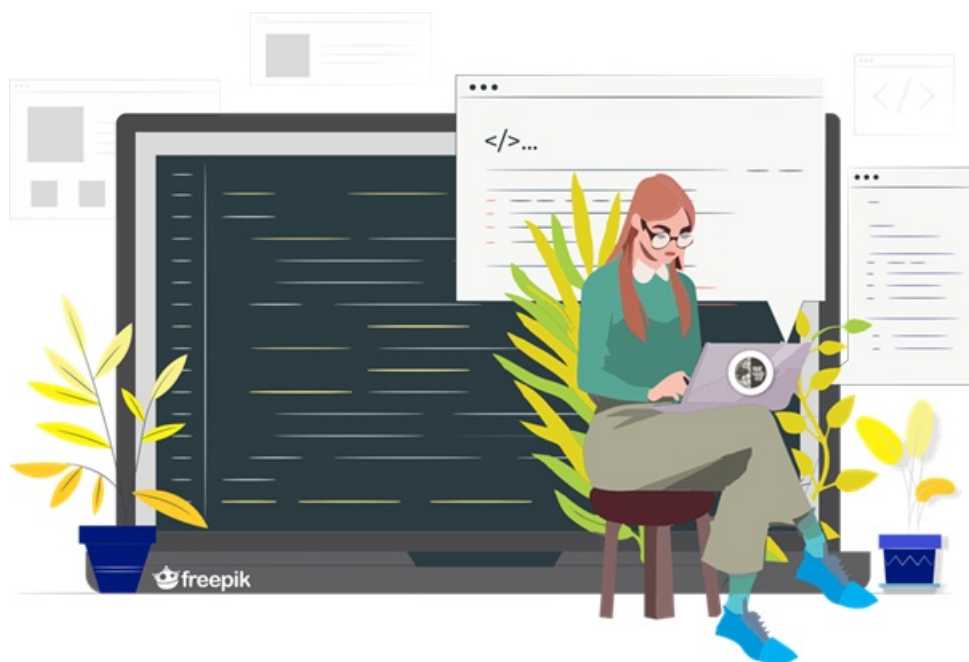


Figura 1.30: Imagen final del circuito simplificado.



Laboratorio 2

Capítulo 1

1.4. Ejercicios del Laboratorio 2

A continuación empezamos con los ejercicios del laboratorio 2 de las prácticas de experimentación del capítulo 1.

1.4.1. Sección 1 del laboratorio 2.

Enunciado 1: A continuación, representa con notación lógica el siguiente argumento.

“El programa está bien si y sólo si no tiene errores de compilación. Si no tiene errores de lógica y no tiene errores de compilación, entonces el programa está bien y los resultados son satisfactorios. Por lo tanto, si tiene errores de compilación o tiene errores de lógica, entonces el programa no está bien y los resultados no son satisfactorios.”

1. Identificación de variables y notación lógica, ver Tabla 1.60

Variables proposicionales y notación

Variables definidas

P: El programa está bien.

Q: El programa tiene errores de compilación.

R: El programa tiene errores de lógica.

S: Los resultados son satisfactorios.

Notación lógica del argumento

Tabla 1.60: Traducción de premisas y conclusión del enunciado 1

Parte del enunciado	Notación lógica
Premisa 1: El programa está bien si y sólo si no tiene errores de compilación.	$P \leftrightarrow \neg Q$
Premisa 2: Si no tiene errores de lógica y no tiene errores de compilación, entonces el programa está bien y los resultados son satisfactorios.	$(\neg R \wedge \neg Q) \rightarrow (P \wedge S)$
Conclusión: Si tiene errores de compilación o tiene errores de lógica, entonces el programa no está bien y los resultados no son satisfactorios.	$(Q \vee R) \rightarrow (\neg P \wedge \neg S)$

Proposición compuesta total (T)

El argumento completo se formula como una implicación [Premisas] $\rightarrow$ [Conclusión]: $T = [(P \leftrightarrow \neg Q) \wedge ((\neg R \wedge \neg Q) \rightarrow (P \wedge S))] \rightarrow [(Q \vee R) \rightarrow (\neg P \wedge \neg S)]$

2. Tabla de verdad y clasificación, ver Tabla 1.61

2. Tabla de verdad y clasificación

Evaluamos: $T = [(P \leftrightarrow \neg Q) \wedge ((\neg R \wedge \neg Q) \rightarrow (P \wedge S))] \rightarrow [(Q \vee R) \rightarrow (\neg P \wedge \neg S)]$

Tabla 1.61: Evaluación completa de la proposición de 4 variables del enunciado 1

Entradas				Proposiciones y Evaluación									
P	Q	R	S	¬Q	P ↔ ¬Q	¬R	¬R ∧ ¬Q	P ∧ S	P ₂	P ₁ ∧ P ₂	Q ∨ R	¬P ∧ ¬S	T
F	F	F	F	V	F	V	V	F	F	F	F	V	V
F	F	F	V	V	F	V	V	F	F	F	F	V	V
F	F	V	F	V	F	F	F	F	V	F	V	F	V
F	F	V	V	V	F	F	F	F	V	F	V	F	V
F	V	F	F	F	V	V	F	F	V	V	V	V	V
F	V	F	V	F	V	V	F	F	V	V	V	F	F
F	V	V	F	F	V	F	F	F	V	V	V	V	V
F	V	V	V	F	V	F	F	F	V	V	V	F	F
V	F	F	F	V	V	V	V	F	F	F	F	F	V
V	F	F	V	V	V	V	V	V	V	V	F	F	V
V	F	V	F	V	V	F	F	F	V	V	V	F	V
V	F	V	V	V	V	F	F	V	V	V	V	F	V
V	V	F	F	F	F	V	F	F	V	F	V	F	V
V	V	F	V	F	F	V	F	V	V	F	V	F	V
V	V	V	F	F	F	F	F	F	V	F	V	F	V
V	V	V	V	F	F	F	F	V	V	F	V	F	V

Clasificación del resultado: La columna de la proposición compuesta total (T) muestra tanto valores (V) como (F) (filas 6 y 8). Por ende, la proposición es una **Contingencia**.

Enunciado 2: A continuación, representa con notación lógica el siguiente argumento.

“Si se realiza un buen diseño de la base de datos y se hace una buena programación, entonces se accederá rápidamente la información. Si no se hace buena programación, entonces toma mucho tiempo corregir el programa. Por lo tanto, si no se accede rápidamente la información y toma mucho tiempo corregir el programa, entonces no se ha realizado un buen diseño de la base de datos.”

1. Identificación de variables y notación lógica, ver Tabla 1.62

Variables proposicionales y notación

Variables definidas

- P:** Se realiza un buen diseño de la base de datos.
- Q:** Se hace una buena programación.
- R:** Se accederá rápidamente la información.
- S:** Toma mucho tiempo corregir el programa.

Tabla 1.62: Traducción de premisas y conclusión del enunciado 2

Parte del enunciado	Notación lógica
Premisa 1: Si se realiza un buen diseño de la base de datos y se hace una buena programación, entonces se accederá rápidamente la información.	$(P \wedge Q) \rightarrow R$
Premisa 2: Si no se hace buena programación, entonces toma mucho tiempo corregir el programa.	$\neg Q \rightarrow S$
Conclusión: Si no se accede rápidamente la información y toma mucho tiempo corregir el programa, entonces no se ha realizado un buen diseño de la base de datos.	$(\neg R \wedge S) \rightarrow \neg P$

Proposición $T = [(P \wedge Q) \rightarrow R] \wedge (\neg Q \rightarrow S) \rightarrow [(\neg R \wedge S) \rightarrow \neg P]$

2. Tabla de verdad y clasificación, ver Tabla 1.63

2. Tabla de verdad y clasificación

Evaluamos $T = [((P \wedge Q) \rightarrow R) \wedge (\neg Q \rightarrow S)] \rightarrow [(\neg R \wedge S) \rightarrow \neg P]$

Tabla 1.63: Evaluación completa de la proposición de 4 variables del enunciado 2

Entradas				Proposiciones y Evaluación							
P	Q	R	S	$P \wedge Q$	P_1	$\neg Q$	P_2	$P_1 \wedge P_2$	$\neg R \wedge S$	$\neg P$	T
F	F	F	F	F	V	V	F	F	F	V	V
F	F	F	V	F	V	V	V	V	V	V	V
F	F	V	F	F	V	V	F	F	F	V	V
F	F	V	V	F	V	V	V	V	F	V	V
F	V	F	F	F	V	F	V	V	F	V	V
F	V	F	V	F	V	F	V	V	V	V	V
F	V	V	F	F	V	F	V	V	F	V	V
F	V	V	V	F	V	F	V	V	F	V	V
V	F	F	F	F	V	V	F	F	F	F	V
V	F	F	V	F	V	V	V	V	V	F	F
V	F	V	F	F	V	V	F	F	F	F	V
V	F	V	V	F	V	V	V	V	F	F	V
V	V	F	F	V	F	F	V	F	F	F	V
V	V	F	V	V	F	F	V	F	V	F	V
V	V	V	F	V	V	F	V	V	F	F	V
V	V	V	V	V	V	F	V	V	F	F	V

Resultado: La columna (T) contiene un valor (F) en la fila 10 (VFFV). En todos los demás casos es (V). Por lo tanto, la proposición se clasifica como una **Contingencia**.

Enunciado 3: A continuación, representa con notación lógica el siguiente argumento.

“Si tengo conocimientos de computación y domino el inglés, entonces no tendré problemas para encontrar trabajo. Si tengo problemas para encontrar trabajo, entonces tengo más de 40 años o no me preparé lo suficiente. Por lo tanto, si me preparo lo suficiente y no tengo más de 40 años y domino el inglés, entonces no tendré problemas para encontrar trabajo.”

1. Identificación de variables y notación lógica, ver Tabla 1.64

Variables proposicionales y notación

- P:** Tengo conocimientos de computación.
- Q:** Domino el inglés.
- R:** Tendré problemas para encontrar trabajo.
- S:** Tengo más de 40 años.
- T:** Me preparé lo suficiente.

Notación: $W = [((P \wedge Q) \rightarrow \neg R) \wedge (R \rightarrow (S \vee \neg T))] \rightarrow [(T \wedge \neg S \wedge Q) \rightarrow \neg R]$

Tabla 1.64: Traducción de premisas y conclusión del enunciado 3

Parte del enunciado	Notación lógica
Premisa 1: Si tengo conocimientos de computación y domino el inglés, entonces no tendré problemas para encontrar trabajo.	$(P \wedge Q) \rightarrow \neg R$
Premisa 2: Si tengo problemas para encontrar trabajo, entonces tengo más de 40 años o no me preparé lo suficiente.	$R \rightarrow (S \vee \neg T)$
Conclusión: Si me preparo lo suficiente y no tengo más de 40 años y domino el inglés, entonces no tendré problemas para encontrar trabajo.	$(T \wedge \neg S \wedge Q) \rightarrow \neg R$

2. Tabla de verdad y clasificación, ver Tabla 1.65

2. Tabla de verdad y clasificación (5 Variables: 32 Filas)

Evaluamos $W = [((P \wedge Q) \rightarrow \neg R) \wedge (R \rightarrow (S \vee \neg T))] \rightarrow [(T \wedge \neg S \wedge Q) \rightarrow \neg R]$

Tabla 1.65: Evaluación completa de la proposición de 5 variables

Entradas					Proposiciones y Evaluación								
P	Q	R	S	T	$\neg R$	$P \wedge Q$	P_1	$\neg T$	$S \vee \neg T$	P_2	$P_1 \wedge P_2$	C	W
F	F	F	F	F	V	F	V	V	V	V	V	V	V
F	F	F	F	V	V	F	V	F	V	V	V	V	V
F	F	F	V	F	V	F	V	V	V	V	V	V	V
F	F	F	V	V	V	F	V	F	V	V	V	V	V
F	F	V	F	F	F	F	V	F	F	F	F	V	V
F	F	V	F	V	F	F	V	V	V	V	V	V	V
F	F	V	V	F	F	F	V	V	V	V	V	V	V
F	F	V	V	V	V	F	V	V	V	V	V	V	V
F	V	F	F	F	V	F	V	V	V	V	V	V	V
F	V	F	F	V	V	F	V	F	F	V	V	V	V
F	V	F	V	F	V	F	V	V	V	V	V	V	V
F	V	F	V	V	V	F	V	F	V	V	V	V	V
F	V	V	F	F	V	F	V	V	V	V	V	V	V
F	V	V	F	V	V	F	V	F	F	F	F	F	V
F	V	V	V	F	V	F	V	V	V	V	V	V	V
F	V	V	V	V	V	F	V	V	V	V	V	V	V
V	F	F	F	F	V	F	V	V	V	V	V	V	V
V	F	F	F	V	V	F	V	F	F	V	V	V	V
V	F	F	V	F	V	F	V	V	V	V	V	V	V
V	F	F	V	V	V	F	V	F	V	V	V	V	V
V	F	V	F	F	F	F	V	V	V	V	V	V	V
V	F	V	F	V	V	F	V	F	F	F	F	V	V
V	F	V	V	F	V	F	V	V	V	V	V	V	V
V	F	V	V	V	V	F	V	F	V	V	V	V	V
V	V	F	F	F	V	V	V	V	V	V	V	F	V
V	V	F	V	F	V	V	V	V	V	V	V	V	V
V	V	F	V	V	V	V	V	F	V	V	V	V	V
V	V	V	F	F	V	V	F	V	V	V	F	V	V
V	V	V	F	V	V	V	F	F	F	F	F	F	V
V	V	V	V	F	V	V	F	V	V	V	F	V	V
V	V	V	V	V	V	V	F	F	V	V	F	V	V

Resultado: La columna (**W**) contiene un valor (**F**), fila 26 (VVFFV). En el resto de los casos es (**V**). Por lo tanto, la proposición se clasifica como una **Contingencia**.

Enunciado 4: A continuación, representa con notación lógica el siguiente argumento.

“Si hace buen tiempo, vamos a la playa. Si vamos a la playa o hay mucha gente, entonces no nos quedamos en casa. Por lo tanto, si no vamos a la playa y nos quedamos en casa, entonces no hace buen tiempo.”

1. Identificación de variables y notación lógica, ver Tabla 1.66

Variables proposicionales y notación

Variables, P: Hace buen tiempo. **Q:** Vamos a la playa. y **R:** Nos quedamos en casa.

Notación $T = [(P \rightarrow Q) \wedge (Q \rightarrow \neg R)] \rightarrow [(\neg Q \wedge R) \rightarrow \neg P]$

Tabla 1.66: Traducción de premisas y conclusión del enunciado 4

Parte del enunciado	Notación lógica
Premisa 1: Si hace buen tiempo, vamos a la playa.	$P \rightarrow Q$
Premisa 2: Si vamos a la playa, entonces no nos quedamos en casa.	$Q \rightarrow \neg R$
Conclusión: Si no vamos a la playa y nos quedamos en casa, entonces no hace buen tiempo.	$(\neg Q \wedge R) \rightarrow \neg P$

2. Tabla de verdad y clasificación, ver Tabla 1.67

2. Tabla de verdad y clasificación (3 Variables: 8 Filas)

Evaluamos $T = [(P \rightarrow Q) \wedge (Q \rightarrow \neg R)] \rightarrow [(\neg Q \wedge R) \rightarrow \neg P]$

Tabla 1.67: Evaluación completa de la proposición de 3 variables

Entradas			Proposiciones y Evaluación						
P	Q	R	$P \rightarrow Q$	$\neg R$	$Q \rightarrow \neg R$	$P_1 \wedge P_2$	$\neg Q \wedge R$	$\neg P$	T
F	F	F	V	V	V	V	F	V	V
F	F	V	V	F	V	V	V	V	V
F	V	F	V	V	V	V	F	V	V
F	V	V	V	F	F	F	F	V	V
V	F	F	F	V	V	F	F	F	V
V	F	V	F	F	V	F	V	F	V
V	V	F	V	V	V	V	F	F	V
V	V	V	V	F	F	F	F	F	V

Clasificación del resultado: La columna de la proposición compuesta total (**T**) contiene **solamente** valores *Verdaderos* (V). Por lo tanto, la proposición se clasifica como una **Tautología**.

Enunciado 5: A continuación, representa con notación lógica el siguiente argumento.

“Si el sistema está en línea, los usuarios pueden acceder. Por lo tanto, el sistema no está en línea o los usuarios pueden acceder.”

Análisis lógico completo (2 variables), ver Tabla 1.68

Análisis de la Proposición (Equivalencia Lógica)

1. Variables y notación

P: El sistema está en línea.

Q: Los usuarios pueden acceder.

Notación $T = (P \rightarrow Q) \rightarrow (\neg P \vee Q)$

2. Tabla de Verdad y clasificación

Tabla 1.68: Evaluación de $T = (P \rightarrow Q) \rightarrow (\neg P \vee Q)$

Entradas		Proposiciones y Evaluación			
P	Q	$P \rightarrow Q$	$\neg P$	$\neg P \vee Q$	T
V	V	V	F	V	V
V	F	F	F	F	V
F	V	V	V	V	V
F	F	V	V	V	V

Clasificación del resultado: La columna de la proposición compuesta total (**T**) contiene **solamente** valores *Verdaderos* (V). Por lo tanto, la proposición se clasifica como una **Tautología**.

Enunciado 6: A continuación, representa con notación lógica el siguiente argumento.

“Si invierto en criptomonedas y la economía es estable, gano dinero. Si no invierto en criptomonedas, la economía no es estable. Por lo tanto, si gano dinero, la economía es estable.”

Análisis lógico completo (3 variables), ver Tabla 1.69 y Tabla 1.70

Análisis de la Proposición (Contingencia)

1. Variables y notación

P: Invierto en criptomonedas.

Q: La economía es estable.

R: Gano dinero.

Notación $T = [((P \wedge Q) \rightarrow R) \wedge (\neg P \rightarrow \neg Q)] \rightarrow (R \rightarrow Q)$

Tabla 1.69: Traducción de premisas y conclusión del enunciado 6

Parte del enunciado	Notación lógica
Premisa 1: Si invierto en criptomonedas y la economía es estable, gano dinero.	$(P \wedge Q) \rightarrow R$
Premisa 2: Si no invierto en criptomonedas, la economía no es estable.	$\neg P \rightarrow \neg Q$
Conclusión: Si gano dinero, la economía es estable.	$R \rightarrow Q$

2. Tabla de Verdad y clasificación

Tabla 1.70: Evaluación de $T = [((P \wedge Q) \rightarrow R) \wedge (\neg P \rightarrow \neg Q)] \rightarrow (R \rightarrow Q)$

Entradas			Proposiciones y Evaluación							
P	Q	R	$P \wedge Q$	P_1	$\neg P$	$\neg Q$	P_2	$P_1 \wedge P_2$	$R \rightarrow Q$	T
V	V	V	V	V	F	F	V	V	V	V
V	V	F	V	F	F	F	V	F	V	V
V	F	V	F	V	F	V	V	V	F	F
V	F	F	F	V	F	V	V	V	V	V
F	V	V	F	V	V	F	F	F	V	V
F	V	F	F	V	V	F	F	F	V	V
F	F	V	F	V	V	V	V	V	F	F
F	F	F	F	V	V	V	V	V	V	V

Clasificación del resultado: La columna de la proposición compuesta total (T) contiene valores *Verdaderos* (V) y *Falsos* (F) (en las filas 3 y 7). Por lo tanto, la proposición se clasifica como una **Contingencia**.

Enunciado 7: A continuación, representa con notación lógica el siguiente argumento.

“Si un estudiante asiste a todas las clases y estudia para el examen, entonces aprueba el curso. Si el estudiante aprueba el curso, obtiene su título. El estudiante no asiste a todas las clases o no obtiene su título. Por lo tanto, el estudiante no aprueba el curso o no estudia para el examen.”

Análisis lógico completo (4 variables), ver Tabla 1.71 y Tabla 1.72

Análisis de la Proposición (Contingencia)

1. Variables y notación

- P:** El estudiante asiste a todas las clases.
Q: El estudiante estudia para el examen.
R: El estudiante aprueba el curso.
S: El estudiante obtiene su título.

Tabla 1.71: Traducción de premisas y conclusión del enunciado 7

Parte del enunciado	Notación lógica
Premisa 1: Si asiste a clases y estudia, entonces aprueba el curso.	$(P \wedge Q) \rightarrow R$
Premisa 2: Si el estudiante aprueba el curso, obtiene su título.	$R \rightarrow S$
Premisa 3: El estudiante no asiste a clases o no obtiene su título.	$\neg P \vee \neg S$
Conclusión: El estudiante no aprueba el curso o no estudia para el examen.	$\neg R \vee \neg Q$

Notación $T = [((P \wedge Q) \rightarrow R) \wedge (R \rightarrow S) \wedge (\neg P \vee \neg S)] \rightarrow (\neg R \vee \neg Q)$

2. Tabla de Verdad y clasificación

Tabla 1.72: Evaluación de $T = [((P \wedge Q) \rightarrow R) \wedge (R \rightarrow S) \wedge (\neg P \vee \neg S)] \rightarrow (\neg R \vee \neg Q)$

Entradas				Proposiciones y Evaluación								
P	Q	R	S	$P \wedge Q$	P ₁	P ₂	$\neg P$	$\neg S$	P ₃	PREMISAS	CONCL.	T
F	F	F	F	F	V	V	V	V	V	V	V	V
F	F	F	V	F	V	V	V	F	V	V	V	V
F	F	V	F	F	V	F	V	V	V	F	F	V
F	F	V	V	F	V	V	V	F	V	V	F	F
F	V	F	F	F	V	V	V	V	V	V	V	V
F	V	F	V	F	V	V	V	F	V	V	V	V
F	V	V	F	F	V	F	V	V	V	F	F	V
F	V	V	V	F	V	V	V	F	V	V	F	F
V	F	F	F	F	V	V	F	V	F	F	V	V
V	F	F	V	F	V	V	F	F	F	F	V	V
V	F	V	F	F	V	F	F	V	F	F	F	V
V	F	V	V	F	V	V	F	F	F	F	F	V
V	V	F	F	V	F	V	F	V	F	F	V	V
V	V	F	V	V	F	V	F	F	F	F	V	V
V	V	V	F	V	V	F	F	V	F	F	F	V
V	V	V	V	V	V	V	F	F	F	F	F	V

Clasificación del resultado: La columna de la proposición compuesta total (T) contiene valores *Verdaderos* (V) y *Falsos* (F) (en las filas 4, 8, 12 y 16). Por lo tanto, la proposición se clasifica como una **Contingencia**.

Enunciado 8: A continuación, representa con notación lógica el siguiente argumento.

“Si la empresa invierte en I+D, sus productos serán innovadores. Si los productos son innovadores y la publicidad es efectiva, entonces las ventas aumentan. La empresa no invierte en I+D o el mercado es favorable. Si el mercado es favorable y las ventas aumentan, entonces la publicidad es efectiva. Por lo tanto, si las ventas no aumentan, la publicidad no es efectiva y la empresa no invierte en I+D.”

Análisis lógico completo (5 variables), ver Tabla 1.73 y Tabla 1.74

Análisis de la Proposición (5 Variables: 32 Filas)

La variables son: **P:** La empresa invierte en I+D. **Q:** Los productos son innovadores. **R:** La publicidad es efectiva. **S:** Las ventas aumentan. **T:** El mercado es favorable.

Tabla 1.73: Traducción de premisas y conclusión del enunciado 8

Parte del enunciado	Notación lógica
Premisa 1: Si la empresa invierte en I+D, sus productos serán innovadores.	$P \rightarrow Q$
Premisa 2: Si los productos son innovadores y la publicidad es efectiva, entonces las ventas aumentan.	$(Q \wedge R) \rightarrow S$
Premisa 3: La empresa no invierte en I+D o el mercado es favorable.	$\neg P \vee T$
Premisa 4: Si el mercado es favorable y las ventas aumentan, entonces la publicidad es efectiva.	$(T \wedge S) \rightarrow R$
Conclusión: Si las ventas no aumentan, la publicidad no es efectiva y la empresa no invierte en I+D.	$\neg S \rightarrow (\neg R \wedge \neg P)$

$$W = [(P \rightarrow Q) \wedge ((Q \wedge R) \rightarrow S) \wedge (\neg P \vee T) \wedge ((T \wedge S) \rightarrow R)] \rightarrow (\neg S \rightarrow (\neg R \wedge \neg P))$$

Tabla 1.74: Evaluación enunciado 8 - $W = [P_1 \wedge P_2 \wedge P_3 \wedge P_4] \rightarrow \text{Conclusión}$

Entradas					Proposiciones y Evaluación												
P	Q	R	S	T	P ₁	Q ∧ R	P ₂	P ₃	T ∧ S	P ₄	PREMISAS	¬S	CONCL.	W			
F	F	F	F	F	V	F	V	V	F	V	V	V	V	V			
F	F	F	F	V	V	F	V	V	F	V	V	V	V	V			
F	F	F	V	F	V	F	V	V	F	V	V	F	V	V			
F	F	F	V	V	V	F	V	V	V	F	F	F	V	V			
F	F	V	F	F	V	F	V	V	F	V	V	V	V	V			
F	F	V	F	V	V	F	V	V	F	V	V	V	V	V			
F	F	V	V	F	V	F	V	V	F	V	V	F	V	V			
F	F	V	V	V	V	F	V	V	V	V	V	F	V	V			
F	V	F	F	F	V	F	V	V	F	V	V	V	V	V			
F	V	F	F	V	V	F	V	V	F	V	V	V	V	V			
F	V	F	V	F	V	F	V	V	V	F	F	F	V	V			
F	V	F	V	V	V	F	V	V	V	V	V	F	V	V			
F	V	V	F	F	V	V	F	V	F	V	F	V	V	V			
F	V	V	F	V	V	V	F	V	F	V	F	V	V	V			
F	V	V	V	F	V	V	V	V	F	V	V	F	V	V			
F	V	V	V	V	V	V	V	V	V	V	V	F	V	V			
V	F	F	F	F	F	F	V	V	F	V	F	V	V	V			
V	F	F	F	V	F	F	V	V	F	V	F	V	V	V			
V	F	F	V	F	F	F	V	V	F	V	F	F	V	V			
V	F	F	V	V	F	F	V	V	V	F	F	F	V	V			
V	F	V	F	F	F	F	V	V	F	V	V	V	V	V			
V	F	V	F	V	V	F	V	V	F	V	V	V	V	V			
V	F	V	V	F	V	F	V	V	V	V	F	F	V	V			
V	F	V	V	V	V	F	V	V	V	V	F	F	V	V			
V	V	F	F	F	V	F	V	F	F	V	F	V	V	V			
V	V	F	F	V	F	F	V	V	F	V	V	V	V	V			
V	V	F	V	F	V	F	V	V	V	V	F	F	V	V			
V	V	F	V	V	V	F	V	V	V	V	F	F	V	V			
V	V	V	F	F	V	V	V	V	F	V	V	V	V	V			
V	V	V	F	V	V	V	V	V	F	V	V	V	V	V			
V	V	V	V	F	V	V	V	V	F	V	V	V	V	V			
V	V	V	V	V	V	V	V	V	V	V	V	V	V	V			

Clasificación del resultado: La columna de la proposición compuesta total (**W**) contiene *solamente* valores *Verdaderos* (V) en las 32 filas. Por lo tanto, la proposición se clasifica como una **Tautología**.

Enunciado 9: A continuación, representa con notación lógica el siguiente argumento.

“La empresa implementa IA generativa (P) y la calidad del código (Q) es alta. La satisfacción del cliente (S) aumenta si la interfaz de usuario (R) se mejora o la calidad del código no es alta. Si el gasto en capacitación (T) es alto, entonces la empresa implementa IA generativa. Si la satisfacción del cliente aumenta, entonces el gasto en capacitación es alto. Por lo tanto, la interfaz de usuario se mejora o la satisfacción del cliente aumenta.”

Análisis lógico completo (5 variables), ver Tabla 1.75 y Tabla 1.76

Análisis de la Proposición (5 Variables: 32 Filas)

Variables. **P:** La empresa implementa IA generativa. **Q:** La calidad del código es alta. **R:** La interfaz de usuario se mejora. **S:** La satisfacción del cliente aumenta. **T:** El gasto en capacitación es alto.

Tabla 1.75: Traducción de premisas y conclusión del enunciado 9

Parte del enunciado	Notación lógica
Premisa 1 (P_1): La empresa implementa IA generativa y la calidad del código es alta.	$P \wedge Q$
Premisa 2 (P_2): La satisfacción del cliente aumenta si la interfaz de usuario se mejora o la calidad del código no es alta.	$(R \vee \neg Q) \rightarrow S$
Premisa 3 (P_3): Si el gasto en capacitación es alto, entonces la empresa implementa IA generativa.	$T \rightarrow P$
Premisa 4 (P_4): Si la satisfacción del cliente aumenta, entonces el gasto en capacitación es alto.	$S \rightarrow T$
Conclusión: Por lo tanto, la interfaz de usuario se mejora o la satisfacción del cliente aumenta.	$R \vee S$

Notación: $W = [(P \wedge Q) \wedge ((R \vee \neg Q) \rightarrow S) \wedge (T \rightarrow P) \wedge (S \rightarrow T)] \rightarrow (R \vee S)$

Tabla 1.76: Evaluación enunciado 9 - $W = [P_1 \wedge P_2 \wedge P_3 \wedge P_4] \rightarrow$ Conclusión

Entradas					Proposiciones y Evaluación									
P	Q	R	S	T	P ₁	¬Q	R ∨ ¬Q	P ₂	P ₃	P ₄	PREMISAS	R ∨ S	W	
F	F	F	F	F	F	V	V	F	V	V	F	F	V	
F	F	F	F	V	F	V	V	F	F	F	F	F	V	
F	F	F	V	F	F	V	V	V	V	F	F	V	V	
F	F	F	V	V	F	V	V	V	F	V	F	V	V	
F	F	V	F	F	F	V	V	F	V	V	F	V	V	
F	F	V	F	V	F	V	V	V	V	F	F	V	V	
F	F	V	V	F	F	V	V	V	V	V	F	V	V	
F	F	V	V	V	F	V	V	V	F	V	F	V	V	
F	V	F	F	F	F	F	F	V	V	V	F	F	V	
F	V	F	F	V	F	F	F	V	F	F	F	F	V	
F	V	F	V	F	F	F	F	V	V	V	F	V	V	
F	V	F	V	V	F	F	F	V	F	V	F	V	V	
F	V	V	F	F	F	F	V	F	V	V	F	V	V	
F	V	V	F	V	F	F	V	F	F	V	F	V	V	
F	V	V	V	F	F	F	V	V	V	F	F	V	V	
F	V	V	V	V	F	F	V	V	V	V	F	V	V	
V	F	F	F	F	V	V	V	F	V	F	F	F	V	
V	F	F	F	V	F	V	V	V	V	F	F	V	V	
V	F	F	V	F	F	V	V	V	V	V	F	V	V	
V	F	F	V	V	F	V	V	V	V	V	F	V	V	
V	F	V	F	F	F	V	V	F	V	V	F	V	V	
V	F	V	F	V	F	V	V	V	V	F	F	V	V	
V	F	V	V	F	F	V	V	V	V	V	F	V	V	
V	F	V	V	V	F	V	V	V	V	V	V	V	V	
V	V	F	F	F	V	F	F	V	V	V	V	F	F	
V	V	F	F	V	V	F	F	V	V	F	V	F	F	
V	V	F	V	F	V	F	F	V	V	F	V	V	V	
V	V	F	V	V	V	F	V	V	V	V	V	V	V	
V	V	V	F	F	V	F	V	F	V	V	V	V	V	
V	V	V	F	V	V	F	V	V	V	V	V	V	V	
V	V	V	V	F	V	F	V	V	V	F	V	V	V	
V	V	V	V	V	F	V	V	V	V	V	V	V	V	

Clasificación del resultado: La columna (**W**) contiene valores *Verdaderos* (V) y *Falsos* (F) en las 32 filas. Por lo tanto, la proposición se clasifica como una **Contingencia**.

Enunciado 10: Representa con notación lógica el siguiente argumento.

“Si la IA Generativa personaliza el aprendizaje (P), el compromiso del estudiante (Q) aumenta. Los educadores tienen capacitación adecuada (R) y la IA Generativa se implementa en el aula (S). Si el compromiso del estudiante no aumenta y la IA se implementa en el aula, entonces el gasto en tecnología (P) se mantiene bajo o el compromiso del estudiante aumenta. Por lo tanto, si los educadores no tienen capacitación, la IA no personaliza el aprendizaje o el gasto en tecnología se mantiene bajo.”

Análisis lógico completo (4 variables), ver Tabla 1.77 y Tabla 1.78

Análisis de la Proposición (4 Variables: 16 Filas)

Las variables son: **P:** La IA Generativa personaliza el aprendizaje (Gasto en tecnología se mantiene bajo $\neg P$). **Q:** El compromiso del estudiante aumenta. **R:** Los educadores tienen capacitación adecuada. **S:** La IA Generativa se implementa en el aula.

Tabla 1.77: Traducción de premisas y conclusión del enunciado 10

Parte del enunciado	Notación lógica
Premisa 1 (P_1): Si la IA Generativa personaliza el aprendizaje, el compromiso del estudiante aumenta.	$P \rightarrow Q$
Premisa 2 (P_2): Los educadores tienen capacitación adecuada y la IA Generativa se implementa en el aula.	$R \wedge S$
Premisa 3 (P_3): Si el compromiso del estudiante no aumenta y la IA se implementa en el aula, entonces el gasto en tecnología se mantiene bajo o el compromiso del estudiante aumenta.	$(\neg Q \wedge S) \rightarrow (\neg P \vee Q)$
Conclusión: Por lo tanto, si los educadores no tienen capacitación, la IA no personaliza el aprendizaje o el gasto en tecnología se mantiene bajo.	$\neg R \rightarrow (\neg P \vee \neg S)$

Notación: $T = [(P \rightarrow Q) \wedge (R \wedge S) \wedge ((\neg Q \wedge S) \rightarrow (\neg P \vee Q))] \rightarrow (\neg R \rightarrow (\neg P \vee \neg S))$

Tabla 1.78: Evaluación enunciado 10 - $T = [P_1 \wedge P_2 \wedge P_3] \rightarrow$ Conclusión

Entradas				Proposiciones y Evaluación						
P	Q	R	S	P_1	P_2	$\neg Q \wedge S$	P_3	PREMISAS	CONCL.	T
F	F	F	F	V	F	F	V	F	V	V
F	F	F	V	V	F	V	F	F	V	V
F	F	V	F	V	F	F	V	F	V	V
F	F	V	V	V	V	V	V	V	V	V
F	V	F	F	V	F	F	V	F	V	V
F	V	F	V	V	F	F	V	F	V	V
F	V	V	F	V	F	F	V	F	V	V
F	V	V	V	V	V	F	V	V	V	V
V	F	F	F	F	F	F	V	F	V	V
V	F	F	V	F	F	V	V	F	V	V
V	F	V	F	F	F	F	V	F	V	V
V	F	V	V	F	V	V	V	F	V	V
V	V	F	F	V	F	F	V	F	V	V
V	V	F	V	V	F	F	V	F	V	V
V	V	V	F	V	F	F	V	F	V	V
V	V	V	V	V	V	F	V	V	V	V

Clasificación del resultado: La columna de la proposición compuesta total (T) contiene *solamente* valores *Verdaderos* (V) en las 16 filas. Por lo tanto, la proposición se clasifica como una **Tautología**.

Enunciado 11: Representa con notación lógica el siguiente argumento.

“Si el contenido generado por IA es malicioso (P), la desinformación (Q) aumenta. La regulación es estricta (R) si la IA no es transparente. Si la desinformación aumenta y la IA no es transparente, entonces la confianza pública (S) se deteriora. Por lo tanto, si la confianza pública se deteriora, el contenido generado por IA no es malicioso o la regulación no es estricta.”

Análisis lógico completo (4 variables), ver Tabla 1.79 y Tabla 1.80

Análisis de la Proposición (4 Variables: 16 Filas)

La variables son: **P:** El contenido generado por IA es malicioso. **Q:** La desinformación aumenta. **R:** La regulación es estricta. **S:** La IA es transparente ($\neg S$: Confianza pública se deteriora).

Tabla 1.79: Traducción de premisas y conclusión del enunciado 11

Parte del enunciado	Notación lógica
Premisa 1 (P_1): Si el contenido generado por IA es malicioso, la desinformación aumenta.	$P \rightarrow Q$
Premisa 2 (P_2): La regulación es estricta si la IA no es transparente.	$\neg S \rightarrow R$
Premisa 3 (P_3): Si la desinformación aumenta y la IA no es transparente, entonces la confianza pública se deteriora.	$(Q \wedge \neg S) \rightarrow \neg S$
Conclusión: Por lo tanto, si la confianza pública se deteriora, el contenido generado por IA no es malicioso o la regulación no es estricta.	$\neg S \rightarrow (\neg P \vee \neg R)$

Notación: $T = [(P \rightarrow Q) \wedge (\neg S \rightarrow R) \wedge ((Q \wedge \neg S) \rightarrow \neg S)] \rightarrow (\neg S \rightarrow (\neg P \vee \neg R))$

Tabla 1.80: Evaluación enunciado 11 - $T = [P_1 \wedge P_2 \wedge P_3] \rightarrow$ Conclusión

Entradas				Proposiciones y Evaluación							
P	Q	R	S	P_1	$\neg S$	P_2	$Q \wedge \neg S$	P_3	PREMISAS	CONCL.	T
F	F	F	F	V	V	F	F	V	F	F	V
F	F	F	V	V	F	V	F	V	V	V	V
F	F	V	F	V	V	V	F	V	V	V	V
F	F	V	V	V	F	V	F	V	V	V	V
F	V	F	F	V	V	F	V	V	F	F	V
F	V	F	V	V	F	V	F	V	V	V	V
F	V	V	F	V	V	V	V	V	V	V	V
F	V	V	V	V	F	V	F	V	V	V	V
V	F	F	F	F	V	F	F	V	F	F	V
V	F	F	V	F	F	V	F	V	F	V	V
V	F	V	F	F	V	V	F	V	F	V	V
V	F	V	V	F	F	V	F	V	F	V	V
V	V	F	F	V	V	F	V	V	F	F	V
V	V	F	V	V	F	V	F	V	V	V	V
V	V	V	F	V	V	V	V	V	V	V	V
V	V	V	V	V	F	V	F	V	V	V	V

Clasificación del resultado: La columna de la proposición compuesta total (**T**) contiene *solamente* valores *Verdaderos* (*V*) en las 16 filas. Por lo tanto, la proposición se clasifica como una **Tautología**.

1.4.2. Sección 2 del laboratorio 2.

Ejercicio 1: Representa cada una de las notaciones lógicas en forma de enunciados.

Notación lógica: $[(p \vee q) \implies \neg r] \wedge [r \iff s] \implies [(r \wedge s) \implies (\neg p \vee \neg q)]$

Considere las siguientes proposiciones: variable p: Compré un automóvil, variable q: Me levanté más temprano, variable r: Llegué tarde al trabajo, variable s: Fui despedido.

Fase 1: Traducción y representación de la proposición

Argumento y notación lógica (4 variables)

Variables proposicionales:

- p: Compré un automóvil.
- q: Me levanté más temprano.
- r: Llegué tarde al trabajo.
- s: Fui despedido.

Notación lógica:

$$T = [(p \vee q) \implies \neg r] \wedge [r \iff s] \implies [(r \wedge s) \implies (\neg p \vee \neg q)]$$

Estructura del argumento ver Tabla 1.81:

- **Premisa (Pr):** $[(p \vee q) \implies \neg r] \wedge [r \iff s]$
- **Conclusión (C):** $(r \wedge s) \implies (\neg p \vee \neg q)$

Tabla 1.81: Traducción de premisas y conclusión del ejercicio 1

Parte del enunciado	Notación lógica
Premisa 1 (P₁): Si compré un automóvil o me levanté más temprano, entonces no llegué tarde al trabajo.	$(p \vee q) \implies \neg r$
Premisa 2 (P₂): Llegué tarde al trabajo si y solo si fui despedido.	$r \iff s$
Conclusión (C): Si llegué tarde al trabajo y fui despedido, entonces no compré un automóvil o no me levanté más temprano.	$(r \wedge s) \implies (\neg p \vee \neg q)$

Fase 2: Estructura del argumento

Argumento estructurado

Notación lógica a descomponer:

$$T = [(p \vee q) \implies \neg r] \wedge [r \iff s] \implies [(r \wedge s) \implies (\neg p \vee \neg q)]$$

Traducción estructurada

$$T = \underbrace{[(p \vee q) \implies \neg r] \wedge [r \iff s]}_{\text{PREMISA (Pr)}} \implies \underbrace{[(r \wedge s) \implies (\neg p \vee \neg q)]}_{\text{CONCLUSIÓN (C)}}$$

Significado: La verdad de todo el argumento (T) depende de la implicación entre la conjunción de las dos premisas y la conclusión.

Fase 3: Enunciado final

Enunciado final completo del argumento lógico traducido a lenguaje natural

“Si es cierto que si compré un automóvil o me levanté más temprano, entonces **no** llegué tarde al trabajo. Llegué tarde al trabajo si y solo si fui despedido. Por lo tanto, si llegué tarde al trabajo y fui despedido, entonces **no** compré un automóvil o **no** me levanté más temprano.”

Ejercicio 2: Representa cada una de las notaciones lógicas en forma de enunciados.

Notación lógica: $[(\neg p \wedge q) \implies r] \vee [(r \vee s) \iff \neg q] \implies [r \implies (\neg p \wedge s)]$

Considere las siguientes proposiciones: variable p: La vigilancia digital es mínima, variable q: La privacidad del usuario se respeta, variable r: Las leyes son insuficientes, variable s: La seguridad de los datos es robusta.

Fase 1: Traducción y representación de la proposición

Argumento y notación lógica (4 variables)

Variables proposicionales:

- **p:** La vigilancia digital es mínima.
- **q:** La privacidad del usuario se respeta.
- **r:** Las leyes son insuficientes.
- **s:** La seguridad de los datos es robusta.

Notación lógica:

$$T = [(\neg p \wedge q) \implies r] \vee [(r \vee s) \iff \neg q] \implies [r \implies (\neg p \wedge s)]$$

Estructura del argumento ver Tabla 1.82:

- **Premisa (Pr):** $[(\neg p \wedge q) \implies r] \vee [(r \vee s) \iff \neg q]$
- **Conclusión (C):** $[r \implies (\neg p \wedge s)]$

Tabla 1.82: Traducción de premisas y conclusión del ejercicio 2

Parte del enunciado	Notación lógica
Premisa 1 (P₁): Si la vigilancia digital no es mínima y la privacidad del usuario se respeta, entonces las leyes son insuficientes.	$(\neg p \wedge q) \implies r$
Premisa 2 (P₂): Las leyes son insuficientes o la seguridad de los datos es robusta, si y solo si la privacidad del usuario no se respeta.	$(r \vee s) \iff \neg q$
Conclusión (C): Si las leyes son insuficientes, entonces la vigilancia digital no es mínima y la seguridad de los datos es robusta.	$r \implies (\neg p \wedge s)$

Fase 2: Estructura del argumento

Argumento estructurado

Notación lógica a descomponer:

$$T = [(\neg p \wedge q) \implies r] \vee [(r \vee s) \iff \neg q] \implies [r \implies (\neg p \wedge s)]$$

Traducción estructurada

$$\mathbf{T} = \underbrace{[(\neg p \wedge q) \implies r] \vee [(r \vee s) \iff \neg q]}_{\text{PREMISA (Pr)}} \implies \underbrace{[r \implies (\neg p \wedge s)]}_{\text{CONCLUSIÓN (C)}}$$

Significado: La verdad de todo el argumento (T) depende de la implicación entre la disyunción de las dos premisas y la conclusión.

Fase 3: Enunciado final

Enunciado final completo del argumento lógico traducido a lenguaje natural

“Si es cierto que (si la vigilancia digital *no es mínima* y la privacidad del usuario se respeta, entonces las leyes son insuficientes) o (las leyes son insuficientes o la seguridad de los datos es robusta, si y solo si la privacidad del usuario *no se respeta*). Por lo tanto, si las leyes son insuficientes, entonces la vigilancia digital *no es mínima* y la seguridad de los datos es robusta.”

Ejercicio 3: Representa cada una de las notaciones lógicas en forma de enunciados.

Notación: $(\neg p \Rightarrow \neg q) \wedge [\neg r \Rightarrow \neg s] \wedge [(\neg q \vee \neg s) \Rightarrow t] \Rightarrow [\neg t \Rightarrow (p \wedge r)]$

Considere las siguientes proposiciones: variable p: Acelero la motocicleta, variable q: La motocicleta corre, variable r: Freno la motocicleta, variable s: La motocicleta se detiene, variable t: La motocicleta falla.

Fase 1: Traducción y representación de la proposición

Argumento y notación lógica (5 variables)

Variables proposicionales:

- p: Acelero la motocicleta.
- q: La motocicleta corre.
- r: Freno la motocicleta.
- s: La motocicleta se detiene.
- t: La motocicleta falla.

Notación lógica:

$$T = (\neg p \Rightarrow \neg q) \wedge [\neg r \Rightarrow \neg s] \wedge [(\neg q \vee \neg s) \Rightarrow t] \Rightarrow [\neg t \Rightarrow (p \wedge r)]$$

Estructura del argumento ver Tabla 1.83:

- **Premisa (Pr):** $(\neg p \Rightarrow \neg q) \wedge [\neg r \Rightarrow \neg s] \wedge [(\neg q \vee \neg s) \Rightarrow t]$
- **Conclusión (C):** $\neg t \Rightarrow (p \wedge r)$

Tabla 1.83: Traducción de premisas y conclusión del ejercicio 3

Parte del enunciado	Notación lógica
Premisa 1 (P₁): Si no acelero la motocicleta, la motocicleta no corre.	$\neg p \Rightarrow \neg q$
Premisa 2 (P₂): Si no freno la motocicleta, la motocicleta no se detiene.	$\neg r \Rightarrow \neg s$
Premisa 3 (P₃): Si la motocicleta no corre o la motocicleta no se detiene, la motocicleta falla.	$(\neg q \vee \neg s) \Rightarrow t$
Conclusión (C): Si la motocicleta no falla, entonces acelero la motocicleta y freno la motocicleta.	$\neg t \Rightarrow (p \wedge r)$

Fase 2: Estructura del argumento

Argumento estructurado (5 variables)

Notación lógica a descomponer:

$$T = [(\neg p \Rightarrow \neg q) \wedge (\neg r \Rightarrow \neg s) \wedge ((\neg q \vee \neg s) \Rightarrow t)] \Rightarrow [\neg t \Rightarrow (p \wedge r)]$$

Traducción estructurada

$$T = \underbrace{(\neg p \Rightarrow \neg q) \wedge [\neg r \Rightarrow \neg s] \wedge [(\neg q \vee \neg s) \Rightarrow t]}_{\text{PREMISA (Pr)}} \Rightarrow \underbrace{[\neg t \Rightarrow (p \wedge r)]}_{\text{CONCLUSIÓN (C)}}$$

Significado: La verdad de todo el argumento (T) depende de la implicación entre la conjunción de las tres premisas y la conclusión.

Fase 3: Enunciado final

Enunciado final completo del argumento lógico traducido a lenguaje natural

“Si (*no acelero la motocicleta*, la motocicleta *no corre*) y (*si no freno la motocicleta*, la motocicleta *no se detiene*) y (*si la motocicleta no corre o la motocicleta no se detiene*, entonces la motocicleta falla); por lo tanto, si la motocicleta *no falla*, entonces acelero la motocicleta y freno la motocicleta.”

Ejercicio 4: Representa cada una de las notaciones lógicas en forma de enunciados.

Notación lógica: $[(p \wedge \neg q) \Rightarrow r] \wedge [(r \Leftrightarrow s) \vee \neg p] \Rightarrow [\neg s \Rightarrow (\neg q \wedge r)]$

Considere las siguientes proposiciones: variable p: El contenido generado por IA es supervisado, variable q: La desinformación se propaga rápidamente, variable r: La regulación legal es estricta, variable s: Los derechos de autor están protegidos.

Fase 1: Traducción y representación de la proposición

Argumento y notación lógica (4 variables)

Variables proposicionales:

- **p:** El contenido generado por IA es supervisado.
- **q:** La desinformación se propaga rápidamente.
- **r:** La regulación legal es estricta.
- **s:** Los derechos de autor están protegidos.

Notación lógica:

$$T = [(p \wedge \neg q) \Rightarrow r] \wedge [(r \Leftrightarrow s) \vee \neg p] \Rightarrow [\neg s \Rightarrow (\neg q \wedge r)]$$

Estructura del argumento ver Tabla 1.84:

- **Premisa (Pr):** $[(p \wedge \neg q) \Rightarrow r] \wedge [(r \Leftrightarrow s) \vee \neg p]$
- **Conclusión (C):** $\neg s \Rightarrow (\neg q \wedge r)$

Tabla 1.84: Traducción de premisas y conclusión del ejercicio 4

Parte del enunciado	Notación lógica
Premisa 1 (P₁): Si el contenido generado por IA es supervisado y la desinformación no se propaga rápidamente, entonces la regulación legal es estricta.	$(p \wedge \neg q) \Rightarrow r$
Premisa 2 (P₂): La regulación legal es estricta si y solo si los derechos de autor están protegidos, o el contenido generado por IA no es supervisado.	$(r \Leftrightarrow s) \vee \neg p$
Conclusión (C): Si los derechos de autor no están protegidos, entonces la desinformación no se propaga rápidamente y la regulación legal es estricta.	$\neg s \Rightarrow (\neg q \wedge r)$

Fase 2: Estructura del argumento

Argumento estructurado (4 variables)

Notación lógica a descomponer:

$$T = [(p \wedge \neg q) \Rightarrow r] \wedge [(r \Leftrightarrow s) \vee \neg p] \Rightarrow [\neg s \Rightarrow (\neg q \wedge r)]$$

Traducción estructurada

$$T = \underbrace{[(p \wedge \neg q) \Rightarrow r] \wedge [(r \Leftrightarrow s) \vee \neg p]}_{\text{PREMISA (Pr)}} \Rightarrow \underbrace{[\neg s \Rightarrow (\neg q \wedge r)]}_{\text{CONCLUSIÓN (C)}}$$

Significado: La verdad de todo el argumento (T) depende de la implicación entre la conjunción de las dos premisas y la conclusión.

Fase 3: Enunciado final

Enunciado final completo del argumento lógico traducido a lenguaje natural

“Si es cierto que (si el contenido generado por IA es supervisado y la desinformación **no se propaga** rápidamente, **entonces** la regulación legal es estricta) **y** (la regulación legal es estricta **si y solo si** los derechos de autor están protegidos, **o** el contenido generado por IA **no es supervisado**). **Por lo tanto**, si los derechos de autor **no están protegidos**, **entonces** la desinformación **no se propaga** rápidamente y la regulación legal es estricta.”

Ejercicio 5: Representa cada una de las notaciones lógicas en forma de enunciados.

Notación lógica: $[\neg p \implies (q \wedge \neg r)] \vee [r \iff (\neg q \wedge s)] \implies [\neg s \implies (r \vee p)]$

Considere las siguientes proposiciones: variable p: Las sanciones económicas occidentales son efectivas, variable q: La ayuda militar a Ucrania es constante, variable r: Las negociaciones de paz se retoman, variable s: Rusia avanza significativamente en el territorio ucraniano.

Fase 1: Traducción y representación de la proposición

Argumento y notación lógica (4 variables)

Variables proposicionales:

- p: Las sanciones económicas occidentales son efectivas.
- q: La ayuda militar a Ucrania es constante.
- r: Las negociaciones de paz se retoman.
- s: Rusia avanza significativamente en el territorio ucraniano.

Notación lógica:

$$T = [\neg p \implies (q \wedge \neg r)] \vee [r \iff (\neg q \wedge s)] \implies [\neg s \implies (r \vee p)]$$

Estructura del argumento ver Tabla 1.85:

- **Premisa (Pr):** $[\neg p \implies (q \wedge \neg r)] \vee [r \iff (\neg q \wedge s)]$
- **Conclusión (C):** $\neg s \implies (r \vee p)$

Tabla 1.85: Traducción de premisas y conclusión del ejercicio 5

Parte del enunciado	Notación lógica
Premisa 1 (P₁): Si las sanciones económicas no son efectivas, entonces la ayuda militar a Ucrania es constante y las negociaciones de paz no se retoman.	$\neg p \implies (q \wedge \neg r)$
Premisa 2 (P₂): Las negociaciones de paz se retoman si y solo si la ayuda militar a Ucrania no es constante y Rusia avanza significativamente.	$r \iff (\neg q \wedge s)$
Conclusión (C): Si Rusia no avanza significativamente en el territorio ucraniano, entonces las negociaciones de paz se retoman o las sanciones son efectivas.	$\neg s \implies (r \vee p)$

Fase 2: Estructura del argumento

Argumento estructurado (4 variables)

Traducción estructurada

$$T = \underbrace{[\neg p \implies (q \wedge \neg r)] \vee [r \iff (\neg q \wedge s)]}_{\text{PREMISA (Pr)}} \implies \underbrace{[\neg s \implies (r \vee p)]}_{\text{CONCLUSIÓN (C)}}$$

Fase 3: Enunciado final

Enunciado final completo del argumento lógico traducido a lenguaje natural

“Si es cierto que (si las sanciones económicas **no son efectivas**, entonces la ayuda militar a Ucrania es constante y las negociaciones de paz **no se retoman**) o (las negociaciones de paz se retoman si y solo si la ayuda militar a Ucrania **no es constante** y Rusia avanza significativamente en el territorio ucraniano). Por lo tanto, si Rusia **no avanza** significativamente en el territorio ucraniano, entonces las negociaciones de paz se retoman o las sanciones económicas occidentales son efectivas.”

Ejercicio 6: Representa cada una de las notaciones lógicas en forma de enunciados.

Notación lógica: $[(p \wedge \neg q) \Rightarrow r] \vee [\neg r \Leftrightarrow (s \vee q)] \Rightarrow [\neg s \Rightarrow (p \wedge r)]$

Considere las siguientes proposiciones: variable p: Ecuador gana el próximo partido de Eliminatorias, variable q: El equipo se clasifica directamente al Mundial, variable r: El Director Técnico mantiene su puesto, variable s: El goleador histórico anota en el partido.

Fase 1: Traducción y representación de la proposición

Argumento y notación lógica (4 variables)

Variables proposicionales:

- **p:** Ecuador gana el próximo partido de Eliminatorias.
- **q:** El equipo se clasifica directamente al Mundial.
- **r:** El Director Técnico mantiene su puesto.
- **s:** El goleador histórico anota en el partido.

Notación lógica:

$$T = [(p \wedge \neg q) \Rightarrow r] \vee [\neg r \Leftrightarrow (s \vee q)] \Rightarrow [\neg s \Rightarrow (p \wedge r)]$$

Estructura del argumento ver Tabla 1.86:

- **Premisa (Pr):** $[(p \wedge \neg q) \Rightarrow r] \vee [\neg r \Leftrightarrow (s \vee q)]$
- **Conclusión (C):** $\neg s \Rightarrow (p \wedge r)$

Tabla 1.86: Traducción de premisas y conclusión del ejercicio 6

Parte del enunciado	Notación lógica
Premisa 1 (P₁): Si Ecuador gana el próximo partido de Eliminatorias y el equipo no se clasifica directamente al Mundial, entonces el Director Técnico mantiene su puesto.	$(p \wedge \neg q) \Rightarrow r$
Premisa 2 (P₂): El Director Técnico no mantiene su puesto si y solo si el goleador histórico anota en el partido o el equipo se clasifica directamente al Mundial.	$\neg r \Leftrightarrow (s \vee q)$
Conclusión (C): Si el goleador histórico no anota en el partido, entonces Ecuador gana el próximo partido y el Director Técnico mantiene su puesto.	$\neg s \Rightarrow (p \wedge r)$

Fase 2: Estructura del argumento

Argumento estructurado (4 variables)

Traducción estructurada

$$T = \underbrace{[(p \wedge \neg q) \Rightarrow r] \vee [\neg r \Leftrightarrow (s \vee q)]}_{\text{PREMISA (Pr)}} \Rightarrow \underbrace{[\neg s \Rightarrow (p \wedge r)]}_{\text{CONCLUSIÓN (C)}}$$

Fase 3: Enunciado final

Enunciado final completo del argumento lógico traducido a lenguaje natural

“Si es cierto que (si Ecuador gana el próximo partido de Eliminatorias y el equipo **no se clasifica directamente al Mundial, **entonces** el Director Técnico mantiene su puesto) o (el Director Técnico **no mantiene** su puesto **si y solo si** el goleador histórico anota en el partido o el equipo se clasifica directamente al Mundial). Por lo tanto, si el goleador histórico **no anota** en el partido, **entonces** Ecuador gana el próximo partido de Eliminatorias y el Director Técnico mantiene su puesto.”**

1.4.3. Sección 3 del laboratorio 2

En esta sección se va a determinar si las siguientes proposiciones son fórmulas contingentes, contradicciones o tautologías. Para obtener la respuesta debe generarlas con el uso de la calculadora lógica³ que se encuentra en el siguiente enlace: https://sourceforge.net/projects/logiccalculator/?source=typ_redirect. Para usar la calculadora debe tener instalado Java⁴ en su computadora. Sitio de descarga: <https://java.com/es/download/>. Y para ejecutar la calculadora, dar un clic con el botón derecho del ratón en el archivo “LogicCalculator.jar” y seleccionar la opción “Oracle Java Runtime” o “Java™ Platform SE binary” ver Fig. 1.31.

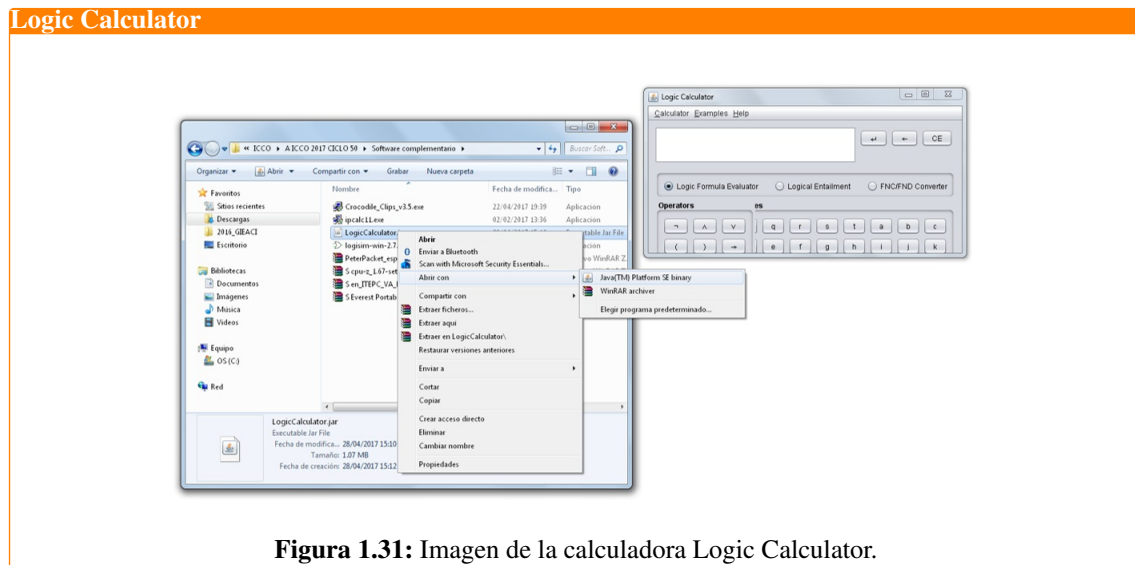


Figura 1.31: Imagen de la calculadora Logic Calculator.

Indicar por qué la proposición es una fórmula contingente, una contradicción o una tautología. Y una vez que se haya generado la tabla de verdad escoger una de las filas de la tabla de verdad y presentar un ejemplo de enunciado. Valor ingresado: $(p \wedge q) \implies (\neg p \vee \neg q)$, ver Fig. 1.32.

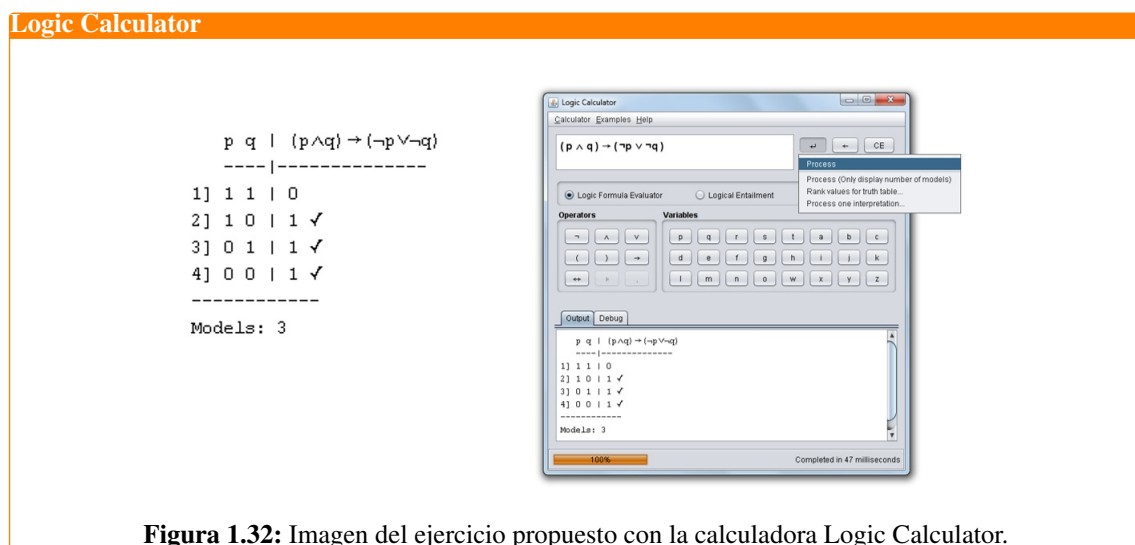


Figura 1.32: Imagen del ejercicio propuesto con la calculadora Logic Calculator.

³Descargar LogicCalculator

⁴Descargar Java

Ejercicio 1: $[\neg p \iff q] \wedge [\neg q \implies (r \vee q)] \wedge p \implies [\neg r \implies \neg p]$

Tabla de verdad del ejercicio 1

Notación lógica: $T = \underbrace{[(\neg p \iff q) \wedge (\neg q \implies (r \vee q)) \wedge p]}_{Pr} \implies \underbrace{[\neg r \implies \neg p]}_C$

Tabla 1.87: Valoración de verdad del argumento, ejercicio 1

p	q	r	$\neg p$	$\neg q$	$\neg r$	$\neg p \iff q$	$r \vee q$	$\neg q \implies (r \vee q)$	Pr	C	T (Pr $\implies$ C)
V	V	V	F	F	F	F	V	V	F	V	V
V	V	F	F	F	V	F	V	V	F	F	V
V	F	V	F	V	F	V	V	V	V	V	V
V	F	F	F	V	V	V	F	F	F	F	V
F	V	V	V	F	F	V	V	V	V	V	V
F	V	F	V	F	V	V	V	V	V	V	V
F	V	V	V	V	F	F	V	V	F	V	V
F	F	F	V	V	V	F	F	F	F	V	V

Resultado: El argumento es una **Tautología**, ya que el valor de verdad final (T) es siempre Verdadero (V) en todas las filas.

Ejercicio 2: $[(q \vee \neg p) \implies r] \wedge [\neg r \implies \neg q] \implies [p \implies \neg q]$

Tabla de verdad del ejercicio 2

Notación lógica: $T = \underbrace{[(q \vee \neg p) \implies r] \wedge [\neg r \implies \neg q]}_{Pr} \implies \underbrace{[p \implies \neg q]}_C$

Tabla 1.88: Valoración de verdad del argumento, ejercicio 2

p	q	r	$\neg p$	$\neg q$	$q \vee \neg p$	$(q \vee \neg p) \implies r$	$\neg r$	$\neg r \implies \neg q$	Pr	T (Pr $\implies$ C)
V	V	V	F	F	V	V	F	V	V	F
V	V	F	F	F	V	F	V	F	F	V
V	F	V	F	V	F	V	F	V	V	V
V	F	F	F	V	F	V	V	V	V	V
F	V	V	V	F	V	V	F	V	V	F
F	V	F	V	F	V	F	V	F	F	V
F	F	V	V	V	V	V	F	V	V	V
F	F	F	V	V	V	F	V	V	F	V

Resultado: El argumento es una **Contingencia**, ya que el valor de verdad final (T) contiene valores Verdaderos (V) y Falsos (F).

Ejercicio 3: $(p \implies q) \wedge (q \implies r) \wedge \neg(p \implies r)$

Tabla de verdad del ejercicio 3

Notación lógica: $T = \underbrace{(p \implies q)}_{P_1} \wedge \underbrace{(q \implies r)}_{P_2} \wedge \underbrace{\neg(p \implies r)}_{P_3}$

Tabla 1.89: Valoración de verdad de la fórmula, ejercicio 3

p	q	r	$p \implies q$	$q \implies r$	$p \implies r$	$\neg(p \implies r)$	$P_1 \wedge P_2$	T
V	V	V	V	V	V	F	V	F
V	V	F	V	F	F	V	F	F
V	F	V	F	V	V	F	F	F
V	F	F	F	V	F	V	F	F
F	V	V	V	V	V	F	V	F
F	V	F	V	F	V	F	F	F
F	F	V	V	V	V	F	V	F
F	F	F	V	V	V	F	V	F

Resultado: El argumento es una **Contradicción**.

Ejercicio 4: $[(p \wedge \neg q) \implies (r \vee s)] \wedge [\neg r \iff (q \wedge \neg s)]$

Tabla de verdad del ejercicio 4

Notación lógica: $T = \underbrace{[(p \wedge \neg q) \implies (r \vee s)]}_{P_1} \wedge \underbrace{[\neg r \iff (q \wedge \neg s)]}_{P_2}$

Tabla 1.90: Valoración de verdad de la fórmula, ejercicio 4

p	q	r	s	$\neg q$	$\neg r$	$\neg s$	$p \wedge \neg q$	$r \vee s$	P_1	$q \wedge \neg s$	P_2	T
V	V	V	V	F	F	F	F	V	V	F	V	V
V	V	V	F	F	F	F	F	V	V	V	F	F
V	V	F	V	F	V	F	F	V	V	F	F	F
V	V	F	F	F	V	V	F	F	V	V	V	V
V	F	V	V	V	F	F	V	V	V	F	V	V
V	F	V	F	V	F	V	V	V	V	F	V	V
V	F	F	V	V	V	F	V	V	V	F	F	F
V	F	F	F	V	V	V	V	F	F	F	F	F
F	V	V	V	F	F	F	F	V	V	F	V	V
F	V	V	F	F	F	V	F	V	V	V	F	F
F	V	F	V	F	V	F	F	V	V	F	F	F
F	V	F	F	F	V	V	F	F	V	V	V	V
F	F	V	V	V	F	F	F	V	V	F	V	V
F	F	V	F	V	F	V	F	V	V	F	V	V
F	F	F	V	V	V	F	F	V	V	F	F	F
F	F	F	F	V	V	V	F	F	V	F	F	F

Resultado: El argumento es una **Contingencia**, ya que el valor de verdad final (T) contiene valores Verdaderos (V) y Falsos (F).

Ejercicio 5: $[(p \implies \neg q) \wedge (\neg r \vee q)] \implies \neg(p \wedge r)$

Tabla de verdad del ejercicio 5

Notación lógica: $T = \underbrace{[(p \implies \neg q) \wedge (\neg r \vee q)]}_{Pr} \implies \underbrace{\neg(p \wedge r)}_C$

Tabla 1.91: Valoración de verdad de la fórmula, ejercicio 5

p	q	r	$\neg q$	$\neg r$	$p \implies \neg q$	$\neg r \vee q$	Pr	$p \wedge r$	C	T
V	V	V	F	F	F	V	F	V	F	V
V	V	F	F	V	F	V	F	F	V	V
V	F	V	V	F	V	F	F	V	F	V
V	F	F	V	V	V	V	V	F	V	V
F	V	V	F	F	V	V	V	F	V	V
F	V	F	F	V	V	V	V	F	V	V
F	F	V	V	F	V	F	F	F	V	V
F	F	F	V	V	V	V	V	F	V	V

Resultado: El argumento es una **Tautología**, ya que el valor de verdad final (T) es siempre Verdadero (V) en todas las filas.

Ejercicio 6: $\neg(p \vee q) \iff (r \wedge s) \wedge [(t \implies u) \wedge \neg t] \implies [(\neg r \wedge p) \vee \neg u]$

Tabla de verdad del ejercicio 6 (64 filas)

Notación lógica: $\mathbf{T} = \underbrace{\neg(p \vee q)}_{P_1} \iff \underbrace{(r \wedge s)}_{P_2} \wedge \underbrace{[(t \implies u) \wedge \neg t]}_{P_2} \implies \underbrace{[(\neg r \wedge p) \vee \neg u]}_C$

Tabla 1.92: Valoración de verdad de la fórmula, ejercicio 6 (con 6 variables)

p	q	r	s	t	u	$\neg(p \vee q)$	$r \wedge s$	P_1	$t \implies u$	$\neg t$	P_2	$\neg r$	$\neg u$	C	T
V	V	V	V	V	V	F	V	F	V	F	F	F	F	F	V
V	V	V	V	V	F	F	V	F	F	F	F	F	V	V	V
V	V	V	V	F	V	F	V	F	V	V	F	F	F	F	V
V	V	V	V	F	F	F	V	F	V	V	F	F	V	V	V
V	V	V	F	V	V	F	F	V	F	F	F	F	F	F	V
V	V	V	F	V	F	F	F	V	F	F	F	F	V	V	V
V	V	V	F	F	V	F	F	V	V	V	F	F	F	F	V
V	V	V	F	F	F	F	F	V	V	V	F	F	V	V	V
V	V	F	V	V	V	F	F	V	V	F	F	V	F	V	V
V	V	F	V	V	F	F	F	V	F	F	F	V	V	V	V
V	V	F	V	F	V	F	F	V	V	V	V	V	F	V	V
V	V	F	V	F	F	F	F	V	V	V	V	V	V	V	V
V	V	F	F	V	V	F	F	V	V	F	F	V	F	V	V
V	V	F	F	V	F	F	F	V	F	F	F	V	V	V	V
V	V	F	F	F	V	F	F	V	V	V	V	V	F	V	V
V	V	F	F	F	F	F	F	V	V	V	V	V	V	V	V
V	F	V	V	V	V	F	V	F	V	F	F	F	F	F	V
V	F	V	V	V	F	F	V	F	F	F	F	F	V	V	V
V	F	V	V	F	V	F	V	F	V	V	V	F	F	F	V
V	F	V	V	F	F	F	V	F	V	V	V	V	V	V	V
V	F	V	F	V	V	F	F	V	V	F	F	F	F	F	V
V	F	V	F	V	F	F	F	V	F	F	F	V	V	V	V
V	F	V	F	F	V	F	F	V	V	V	V	V	V	V	V
V	F	V	F	F	F	F	F	V	V	V	V	V	V	V	V
V	F	F	V	V	V	F	F	V	V	F	F	V	V	V	V
V	F	F	V	V	F	F	F	V	V	V	V	V	V	V	V
V	F	F	V	F	V	F	F	V	V	V	V	V	V	V	V
V	F	F	V	F	F	F	F	V	V	V	V	V	V	V	V
V	F	F	F	V	V	F	F	V	V	V	V	V	V	V	V
V	F	F	F	V	F	F	F	V	V	V	V	V	V	V	V
V	F	F	F	F	V	F	F	V	V	V	V	V	V	V	V
V	F	F	F	F	F	F	F	V	V	V	V	V	V	V	V
F	V	V	V	V	V	F	V	F	V	F	F	F	F	F	V
F	V	V	V	V	F	F	V	F	F	F	F	F	V	V	V
F	V	V	V	F	V	F	V	F	V	V	V	F	F	F	V
F	V	V	V	F	F	F	V	F	V	V	V	V	V	V	V
F	V	V	F	V	V	F	F	V	V	F	F	V	F	F	V
F	V	V	F	V	F	F	F	V	V	V	V	V	V	V	V
F	V	V	F	F	V	F	F	V	V	V	V	V	V	V	V
F	V	V	F	F	F	F	F	V	V	V	V	V	V	V	V
F	V	F	V	V	V	F	F	V	V	F	F	V	V	V	V
F	V	F	V	V	F	F	F	V	V	V	V	V	V	V	V
F	V	F	V	F	V	F	F	V	V	V	V	V	V	V	V
F	V	F	V	F	F	F	F	V	V	V	V	V	V	V	V
F	V	F	F	V	V	F	F	V	V	V	V	V	V	V	V
F	V	F	F	V	F	F	F	V	V	V	V	V	V	V	V
F	V	F	F	F	V	F	F	V	V	V	V	V	V	V	V
F	V	F	F	F	F	F	F	V	V	V	V	V	V	V	V
F	F	V	V	V	V	V	V	V	V	F	F	F	F	F	V
F	F	V	V	V	F	V	V	V	V	F	F	F	V	V	V
F	F	V	V	F	V	V	V	V	V	V	V	F	F	F	V
F	F	V	V	F	F	V	V	V	V	V	V	V	V	V	V
F	F	V	F	V	V	V	F	F	F	F	F	F	V	V	V
F	F	V	F	V	F	V	F	F	F	F	F	F	V	V	V
F	F	V	F	F	V	V	F	F	F	F	F	F	V	V	V
F	F	V	F	F	F	V	F	F	F	F	F	F	V	V	V
F	F	F	V	V	V	V	V	V	V	V	V	V	V	V	V
F	F	F	V	V	F	V	V	V	V	V	V	V	V	V	V
F	F	F	V	F	V	V	V	V	V	V	V	V	V	V	V
F	F	F	V	F	F	V	V	V	V	V	V	V	V	V	V
F	F	F	F	V	V	V	V	V	V	V	V	V	V	V	V
F	F	F	F	V	F	V	V	V	V	V	V	V	V	V	V
F	F	F	F	F	V	V	V	V	V	V	V	V	V	V	V
F	F	F	F	F	F	V	V	V	V	V	V	V	V	V	V
F	F	F	F	F	F	F	V	V	V	V	V	V	V	V	V
F	F	F	F	F	F	F	F	V	V	V	V	V	V	V	V

Resultado (Análisis): Este argumento es una **Contingencia**, ya que el valor de verdad final (T) es Falso en 7 de las 64 combinaciones (las filas marcadas con **FALSO** en la columna T interna).

Ejercicio 7: $[p \Rightarrow q] \Leftrightarrow [\neg q \Rightarrow \neg p]$

Tabla de verdad del ejercicio 7

Notación lógica: $T = \underbrace{[p \Rightarrow q]}_P \Leftrightarrow \underbrace{[\neg q \Rightarrow \neg p]}_C$

Tabla 1.93: Valoración de verdad de la fórmula, ejercicio 7

p	q	$\neg p$	$\neg q$	$p \Rightarrow q$	$\neg q \Rightarrow \neg p$	T
V	V	F	F	V	V	V
V	F	F	V	F	F	V
F	V	V	F	V	V	V
F	F	V	V	V	V	V

Resultado: El argumento es una **Tautología**.

Ejercicio 8: $[p \Rightarrow (q \wedge r)] \wedge [\neg p \Rightarrow \neg q] \Rightarrow [\neg r \Rightarrow p]$

Tabla de verdad del ejercicio 8

Notación lógica: $T = \underbrace{[p \Rightarrow (q \wedge r)] \wedge [\neg p \Rightarrow \neg q]}_{Pr} \Rightarrow \underbrace{[\neg r \Rightarrow p]}_C$

Tabla 1.94: Valoración de verdad de la fórmula, ejercicio 8

p	q	r	$\neg p$	$\neg q$	$\neg r$	$q \wedge r$	$p \Rightarrow (q \wedge r)$	$\neg p \Rightarrow \neg q$	Pr	C	T
V	V	V	F	F	F	V	V	V	V	V	V
V	V	F	F	F	V	F	F	V	F	V	V
V	F	V	F	V	F	F	F	V	F	V	V
V	F	F	F	V	V	F	F	V	F	V	V
F	V	V	V	F	F	V	V	F	F	V	V
F	V	F	V	F	V	F	V	F	F	F	V
F	F	V	V	V	F	F	V	V	V	V	V
F	F	F	V	V	V	F	V	V	V	F	F

Resultado: El argumento es una **Contingencia**.

Ejercicio 9: $[(p \vee q) \Rightarrow (r \wedge \neg s)] \wedge [\neg r \Rightarrow p] \Rightarrow [s \Rightarrow \neg q]$

Tabla de verdad del ejercicio 9

Notación lógica: $T = \underbrace{[(p \vee q) \Rightarrow (r \wedge \neg s)] \wedge [\neg r \Rightarrow p]}_{Pr} \Rightarrow \underbrace{[s \Rightarrow \neg q]}_C$

Tabla 1.95: Valoración de verdad de la fórmula, ejercicio 9

p	q	r	s	$\neg q$	$\neg r$	$\neg s$	$p \vee q$	$r \wedge \neg s$	P ₁	P ₂	Pr	C	T
V	V	V	V	F	F	F	V	F	F	V	F	F	V
V	V	V	F	F	F	V	V	V	V	V	V	V	V
V	V	F	V	F	V	F	V	F	F	V	F	F	V
V	V	F	F	F	V	V	V	F	F	V	F	V	V
V	F	V	V	V	F	F	V	F	F	V	F	V	V
V	F	V	F	V	F	V	V	V	V	V	V	V	V
V	F	F	V	V	V	F	V	F	F	V	F	V	V
V	F	F	F	V	V	V	V	F	F	V	F	V	V
F	V	V	V	F	F	F	V	F	F	V	F	F	V
F	V	V	F	F	F	V	V	V	V	V	V	V	V
F	V	F	V	F	V	F	V	F	F	F	F	F	V
F	V	F	F	F	V	V	V	F	F	F	F	V	V
F	F	V	V	V	F	F	F	F	V	V	V	V	V
F	F	V	F	V	F	V	F	V	V	V	V	V	V
F	F	F	V	V	V	F	F	F	V	F	F	V	V
F	F	F	F	V	V	V	F	F	V	F	F	V	V

Resultado: El argumento es una **Tautología**.

Ejercicio 10: $[(p \wedge \neg q) \vee (r \implies \neg s)] \wedge [\neg p \vee s] \implies [q \iff r]$

Tabla de verdad del ejercicio 10

Notación lógica: $T = \underbrace{[(p \wedge \neg q) \vee (r \implies \neg s)] \wedge [\neg p \vee s]}_{Pr} \implies \underbrace{[q \iff r]}_C$

Tabla 1.96: Valoración de verdad de la fórmula, ejercicio 10

p	q	r	s	$\neg p$	$\neg q$	$\neg s$	$p \wedge \neg q$	$r \implies \neg s$	P ₁	P ₂	Pr	C	T
V	V	V	V	F	F	F	F	F	F	V	F	V	V
V	V	V	F	F	F	V	F	V	V	V	V	V	V
V	V	F	V	F	F	F	F	V	V	V	V	F	F
V	V	F	F	F	F	V	F	V	V	V	V	F	F
V	F	V	V	F	V	F	V	F	V	V	V	F	F
V	F	V	F	F	V	V	V	V	V	V	V	F	F
V	F	F	V	F	V	F	V	V	V	V	V	V	V
V	F	F	F	F	V	V	V	V	V	V	V	V	V
F	V	V	V	V	F	F	F	F	F	V	F	V	V
F	V	V	F	V	F	V	F	V	V	V	V	V	V
F	V	F	V	V	F	F	F	V	V	V	V	F	F
F	V	F	F	V	F	V	F	V	V	V	V	F	F
F	F	V	V	V	V	F	F	F	F	V	F	V	V
F	F	V	F	V	V	V	F	V	V	V	V	F	F
F	F	F	V	V	V	V	F	V	V	V	V	V	V
F	F	F	F	V	V	V	F	V	V	V	V	V	V

Resultado: El argumento es una **Contingencia**.

Ejercicio 11: $[(p \wedge q) \implies r] \wedge [(\neg r \vee p) \implies q] \implies r$

Tabla de verdad del ejercicio 11

Notación lógica: $T = \underbrace{[(p \wedge q) \implies r] \wedge [(\neg r \vee p) \implies q]}_{Pr} \implies \underbrace{r}_C$

Tabla 1.97: Valoración de verdad de la fórmula, ejercicio 11

p	q	r	$\neg r$	$p \wedge q$	P ₁	$\neg r \vee p$	P ₂	Pr	C	T
V	V	V	F	V	V	V	V	V	V	V
V	V	F	V	V	F	V	V	F	F	V
V	F	V	F	F	V	V	F	F	V	V
V	F	F	V	F	V	V	F	F	F	V
F	V	V	F	F	V	F	V	V	V	V
F	V	F	V	F	V	V	V	V	F	F
F	F	V	F	F	V	F	V	V	V	V
F	F	F	V	F	V	V	F	F	F	V

Resultado: El argumento es una **Contingencia**.

Ejercicio 12: $[(p \implies q) \wedge \neg p] \implies \neg q$

Tabla de verdad del ejercicio 12

Notación lógica: $T = \underbrace{[(p \implies q) \wedge \neg p]}_{Pr} \implies \underbrace{\neg q}_C$

Tabla 1.98: Valoración de verdad de la fórmula, ejercicio 12

p	q	$\neg p$	$\neg q$	$p \implies q$	Pr	T
V	V	F	F	V	F	V
V	F	F	V	F	F	V
F	V	V	F	V	V	F
F	F	V	V	V	V	V

Resultado: El argumento es una **Contingencia**.

1.4.4. Sección 4 del laboratorio 2.

Análisis proposicional, ejemplo 1. La proposición a evaluar es: $T = (p \wedge q) \implies (\neg p \vee \neg q)$

1. **Traducción lógica:** La proposición T es lógicamente equivalente a la ley de De Morgan: $(p \wedge q) \implies \neg(p \wedge q)$.
2. **Análisis de resultados:** La proposición es falsa solo cuando su antecedente $(p \wedge q)$ es Verdadero (V) y su consecuente $(\neg p \vee \neg q)$ es Falso (F). Esto ocurre exclusivamente en la primera fila.

p	q	$\neg p$	$\neg q$	$p \wedge q$	$\neg p \vee \neg q$	T
V	V	F	F	V	F	F
V	F	F	V	F	V	V
F	V	V	F	F	V	V
F	F	V	V	F	V	V

Resultado: La proposición es una **Contingencia** porque los valores de verdad se combinan entre Verdadero (V) y Falso (F). **Ejemplo de enunciado:**

Línea	Premisas y Conclusión	Valor
1	p: Tengo novia	1
2	q: Me quiero casar	1
Antecedente	$(p \wedge q)$: (Tengo novia y Me quiero casar)	$1 \wedge 1 = 1$
Consecuente	$(\neg p \vee \neg q)$: (No tengo novia o No Me quiero casar)	$0 \vee 0 = 0$
3	$(p \wedge q) \implies (\neg p \vee \neg q)$: (Tengo novia y Me quiero casar) $\implies$ (No tengo novia o No Me quiero casar)	$1 \implies 0 = 0$

Conclusión del enunciado: La proposición completa es **Falsa** (0), lo cual tiene sentido, pues no es lógico que “Tengo novia y me quiero casar” implique que “No tengo novia o no me quiero casar”.

Derivación de conclusiones: El *Modus Ponendus Ponens* (MPP) tiene la forma: $(A \implies B) \wedge A \implies B$.

■ Argumento 1:

- a. $p \vee q \implies r$
- b. $p \vee q$
- c. r (Por MPP en a y b)

■ Argumento 2:

- a. $\neg p \implies \neg r$
- b. $\neg p$
- c. $\neg r$ (Por MPP en a y b)

■ Argumento 3:

- a. $\neg r$
- b. $\neg r \implies q \wedge p$
- c. $q \wedge p$ (Por MPP en b y a)

Análisis proposicional, ejemplo 2: La proposición a evaluar es un argumento sobre los requisitos para adoptar la IAGen de forma efectiva en un centro educativo: $T = [(\neg p \vee q) \wedge r] \implies (p \implies r)$. Donde: **p**: Los profesores reciben capacitación en IAGen. **q**: La IAGen se integra al currículo de manera efectiva. **r**: La IAGen mejora el rendimiento estudiantil.

1. **Traducción lógica:** La proposición T evalúa si la existencia de *capacitación O implementación* efectiva, junto con la *mejora de rendimiento*, necesariamente implica que *si hay capacitación, hay mejora de rendimiento*.
2. **Análisis de resultados:** La proposición es falsa cuando su antecedente $[(\neg p \vee q) \wedge r]$ es (V) y su consecuente $(p \implies r)$ es (F). En este caso, no ocurre ninguna vez.

p	q	r	$\neg p$	$\neg p \vee q$	Pr	$p \implies r$	T
V	V	V	F	V	V	V	V
V	V	F	F	V	F	F	V
V	F	V	F	F	F	V	V
V	F	F	F	F	F	F	V
F	V	V	V	V	V	V	V
F	V	F	V	V	F	V	V
F	F	V	V	V	V	V	V
F	F	F	V	V	F	V	V

Resultado: La proposición es una **Tautología** porque todos los valores de verdad finales (**T**) son Verdaderos (V). **Ejemplo de enunciado:** (Usando la fila V F V, donde T=V)

Línea	Premisas y Conclusión	Valor
1	p: Los profesores reciben capacitación	1
2	q: IAGen se integra al currículo efectivamente	0
3	r: IAGen mejora el rendimiento	1
Antecedente	$[(\neg p \vee q) \wedge r]$: (No hay capacitación O no hay currículo) $\wedge$ hay mejora	$0 \wedge 1 = 0$
Consecuente	$(p \implies r)$: Si hay capacitación, entonces hay mejora	$1 \implies 1 = 1$
3	Antecedente $\implies$ Consecuente	$0 \implies 1 = 1$

Conclusión del enunciado: La proposición completa es **Verdadera** (1), demostrando la validez del argumento incluso bajo estas condiciones.

Derivación de conclusiones: El *Modus Ponendus Ponens (MPP)* tiene la forma: $(A \implies B) \wedge A \implies B$.

■ Argumento 1:

- a. $(p \wedge q) \implies \neg r$
- b. $p \wedge q$
- c. $\neg r$ (Por MPP en a y b)

■ Argumento 2:

- a. $\neg r \implies \neg p$
- b. $\neg r$
- c. $\neg p$ (Por MPP en a y b)

■ Argumento 3:

- a. $\neg p \vee q$
- b. $(\neg p \vee q) \implies r$
- c. r (Por MPP en b y a)

Análisis proposicional, ejemplo 3. La proposición a evaluar es: $T = [(p \implies q) \wedge \neg q] \implies \neg p$

1. **Traducción lógica:** La proposición T evalúa si Si A implica B, Y B es Falso, necesariamente implica que A también es Falso. (Modus Tollens).
2. **Análisis de resultados:** La proposición es falsa solo cuando su antecedente $[(p \implies q) \wedge \neg q]$ es Verdadero (V) y su consecuente $\neg p$ es Falso (F). Esto no ocurre en ninguna fila.

p	q	$\neg p$	$\neg q$	$p \implies q$	Pr	T
V	V	F	F	V	F	V
V	F	F	V	F	F	V
F	V	V	F	V	F	V
F	F	V	V	V	V	V

Resultado: La proposición es una **Tautología** porque todos los valores de verdad finales (T) son Verdaderos (V). **Ejemplo de enunciado.** Donde: p: Aumenta la temperatura media global. q: Los polos se deshielan.

Línea	Premisas y Conclusión	Valor
1	$\neg q$: Los polos no se deshielan	1
2	$p \implies q$: Si aumenta la temperatura, entonces los polos se deshielan	1
Antecedente	$(p \implies q) \wedge \neg q$: Premisa 2 y Premisa 1	$1 \wedge 1 = 1$
Consecuente	$\neg p$: La temperatura media global no aumenta	1
3	$[(p \implies q) \wedge \neg q] \implies \neg p$: Antecedente $\implies$ Consecuente	$1 \implies 1 = 1$

Conclusión del enunciado: La proposición completa es **Verdadera** (1), confirmando que el Modus Tollens es válido.

Derivación de conclusiones: El *Modus Ponendus Ponens* (MPP) tiene la forma: $(A \implies B) \wedge A \implies B$.

■ Argumento 1:

- a. $p \vee q \implies \neg r$
- b. $p \vee q$
- c. $\neg r$ (Por MPP en a y b)

■ Argumento 2:

- a. $\neg r \implies p$
- b. $\neg r$
- c. p (Por MPP en a y b)

■ Argumento 3:

- a. q
- b. $q \implies p \wedge \neg r$
- c. $p \wedge \neg r$ (Por MPP en b y a)

Análisis proposicional, ejemplo 4. La proposición a evaluar es: $T = [(p \wedge \neg q) \vee (\neg r \implies q)] \implies (r \wedge p)$

1. **Traducción lógica:** La proposición T evalúa si *El chatbot responde rápido Y mal, O si no hay satisfacción implica respuesta correcta*, necesariamente implica que *El cliente queda satisfecho Y la respuesta fue rápida*.
2. **Análisis de resultados:** La proposición es falsa cuando el antecedente $[(p \wedge \neg q) \vee (\neg r \implies q)]$ es Verdadero (V) y el consecuente $(r \wedge p)$ es Falso (F). Esto ocurre en varias filas, confirmando que es una Contingencia.

p	q	r	$\neg q$	$\neg r$	$p \wedge \neg q$	$\neg r \implies q$	Pr	$r \wedge p$	T
V	V	V	F	F	F	V	V	V	V
V	V	F	F	V	F	V	V	F	V
V	F	V	V	F	V	V	V	V	V
V	F	F	V	V	V	F	V	F	F
F	V	V	F	F	F	V	V	F	F
F	V	F	F	V	F	V	V	F	F
F	F	V	V	F	F	V	V	F	F
F	F	F	V	V	F	F	F	F	V

Resultado: La proposición es una **Contingencia**. **Ejemplo de enunciado:** Usando la fila **V F** (Rápido=V, Correcto=F, Satisfecho=F).

Línea	Premisas y Conclusión	Valor
1	p: El chatbot proporciona respuesta rápida	1
2	q: La respuesta es correcta	0
3	r: El cliente queda satisfecho	0
Antecedente	$[(p \wedge \neg q) \vee (\neg r \implies q)]$	$(1 \wedge 1) \vee (1 \implies 0) = 1 \vee 0 = 1$
Consecuente	$(r \wedge p)$: El cliente queda satisfecho Y la respuesta fue rápida	$0 \wedge 1 = 0$
3	Antecedente $\implies$ Consecuente	$1 \implies 0 = 0$

Conclusión del enunciado: La proposición completa es **Falsa** (0), lo cual tiene sentido ya que es posible tener un servicio rápido pero incorrecto (Antecedente V), sin que ello implique que el cliente esté satisfecho (Consecuente F).

Derivación de conclusiones: El *Modus Ponendus Ponens (MPP)* tiene la forma: $(A \implies B) \wedge A \implies B$.

■ Argumento 1:

- $(p \wedge \neg q) \implies \neg r$
- $p \wedge \neg q$
- $\neg r$ (Por MPP en a y b)

■ Argumento 2:

- $\neg r \implies q$
- $\neg r$
- q (Por MPP en a y b)

■ Argumento 3:

- $\neg q \vee \neg p$
- $\neg q \vee \neg p \implies r$
- r (Por MPP en b y a)

1.4.5. Sección 5 del laboratorio 2.

Análisis de inferencia ejercicio 1, Modus Tollendo Tollens (MTT): Si la luz fuera simplemente una onda de movimiento continuo, entonces la luz intensa pudiera siempre causar electrones que escapen con mayor energía que aquella que causaría la luz tenue. La luz intensa no siempre causa electrones que escapen con mayor energía que la que causa la luz tenue.

1. Variables proposicionales:

P: La luz es simplemente una onda de movimiento continuo.

Q: La luz intensa puede siempre causar electrones que escapen con mayor energía que aquella que causaría la luz tenue.

2. Regla de inferencia:

Forma	Nombre
$(P \implies Q) \wedge \neg Q$	Modus Tollendo Tollens (MTT)
$\therefore \neg P$	

3. Simbolización y derivación:

a. **P1** : $P \implies Q$ (Si la luz fuera una onda, causaría Q)

b. **P2** : $\neg Q$ (La luz intensa **no** causa Q)

c. **C** : $\neg P$ (**Por MTT** en a y b)

Conclusión ($\neg P$): La luz **no** es simplemente una onda de movimiento continuo.

Análisis de inferencia ejercicio 2, Modus Tollendo Tollens (MTT): Si el ángulo de un triángulo es mayor que 90 grados, entonces la suma de los otros dos ángulos es menor que 90 grados. La suma de los otros dos ángulos no es menor que 90 grados.

1. Variables proposicionales:

P: El ángulo de un triángulo es mayor que 90 grados.

Q: La suma de los otros dos ángulos es menor que 90 grados.

2. Regla de inferencia:

Forma	Nombre
$(P \implies Q) \wedge \neg Q$	Modus Tollendo Tollens (MTT)
$\therefore \neg P$	

3. Simbolización y derivación:

a. **P1** : $P \implies Q$ (Si un ángulo es $> 90^\circ$, la suma es $< 90^\circ$)

b. **P2** : $\neg Q$ (La suma de los otros dos ángulos **no** es menor que 90°)

c. **C** : $\neg P$ (**Por MTT** en a y b)

Conclusión ($\neg P$): El ángulo de un triángulo **no** es mayor que 90 grados.

Análisis de inferencia ejercicio 3, Modus Tollendo Tollens (MTT): Si llovió la noche anterior, entonces las carreteras fueron lavadas. Las carreteras no han sido lavadas.

1. **Variables proposicionales:**

P: Llovió la noche anterior.

Q: Las carreteras fueron lavadas.

2. **Regla de inferencia:**

Forma	Nombre
$(P \implies Q) \wedge \neg Q$	Modus Tollendo Tollens (MTT)
$\therefore \neg P$	

3. **Simbolización y derivación:**

a. **P1** : $P \implies Q$ (Si llovió, entonces las carreteras fueron lavadas)

b. **P2** : $\neg Q$ (Las carreteras **no** han sido lavadas)

c. **C** : $\neg P$ (**Por MTT** en a y b)

Conclusión ($\neg P$): No llovió la noche anterior.

Análisis de inferencia ejercicio 4: Modus Tollendo Tollens (MTT): Si la sonda espacial aterrizó con éxito, entonces el centro de control recibió una señal de telemetría. El centro de control no ha recibido una señal de telemetría.

1. **Variables proposicionales:**

P: La sonda espacial aterrizó con éxito.

Q: El centro de control recibió una señal de telemetría.

2. **Regla de inferencia:**

Forma	Nombre
$(P \implies Q) \wedge \neg Q$	Modus Tollendo Tollens (MTT)
$\therefore \neg P$	

3. **Simbolización y derivación:**

a. **P1** : $P \implies Q$ (Si la sonda aterrizó, el control recibe telemetría)

b. **P2** : $\neg Q$ (El centro de control **no** ha recibido una señal de telemetría)

c. **C** : $\neg P$ (**Por MTT** en a y b)

Conclusión ($\neg P$): La sonda espacial **no** aterrizó con éxito.

Análisis de inferencia ejercicio 5, Modus Tollendo Tollens (MTT): Si una noticia es verificada por tres fuentes independientes, entonces es legítima. La noticia no es legítima.

1. **Variables proposicionales:**

P: Una noticia es verificada por tres fuentes independientes.

Q: La noticia es legítima (no es una fake news).

2. **Regla de inferencia:**

Forma	Nombre
$(P \implies Q) \wedge \neg Q$	Modus Tollendo Tollens (MTT)
$\therefore \neg P$	

3. **Simbolización y derivación:**

a. **P1** : $P \implies Q$ (Si es verificada, es legítima)

b. **P2** : $\neg Q$ (La noticia **no** es legítima)

c. **C** : $\neg P$ (**Por MTT** en a y b)

Conclusión ($\neg P$): La noticia **no** fue verificada por tres fuentes independientes.

Análisis de inferencia ejercicio 6, Modus Tollendo Tollens (MTT): Si el cometa 3I/ATLAS (detectado por la NASA) alcanza su perihelio, entonces su brillo aparente superará la magnitud +5. Su brillo aparente no superó la magnitud +5.

1. **Variables proposicionales:**

P: El cometa 3I/ATLAS (detectado por la NASA) alcanza su perihelio.

Q: Su brillo aparente superará la magnitud +5.

2. **Regla de inferencia:**

Forma	Nombre
$(P \implies Q) \wedge \neg Q$	Modus Tollendo Tollens (MTT)
$\therefore \neg P$	

3. **Simbolización y derivación:**

a. **P1** : $P \implies Q$ (Si alcanza el perihelio, el brillo es $> +5$)

b. **P2** : $\neg Q$ (Su brillo aparente **no** superó la magnitud +5)

c. **C** : $\neg P$ (**Por MTT** en a y b)

Conclusión ($\neg P$): El cometa 3I/ATLAS (detectado por la NASA) **no** alcanzó su perihelio.

Análisis de inferencia ejercicio 7, Modus Tollendo Tollens (MTT): Si la velocidad de conexión a internet es de al menos 25 Mbps, entonces Netflix transmitirá en calidad Ultra HD (4K). Netflix no está transmitiendo en calidad Ultra HD (4K).

1. **Variables proposicionales:**

P: La velocidad de conexión a internet es de al menos 25 Mbps.

Q: Netflix transmitirá en calidad Ultra HD (4K).

2. **Regla de inferencia:**

Forma	Nombre
$(P \implies Q) \wedge \neg Q$	Modus Tollendo Tollens (MTT)
$\therefore \neg P$	

3. **Simbolización y derivación:**

- a. **P1** : $P \implies Q$ (Si es 25 Mbps o más, transmite en 4K)
- b. **P2** : $\neg Q$ (Netflix **no** está transmitiendo en Ultra HD (4K))
- c. **C** : $\neg P$ (**Por MTT** en a y b)

Conclusión ($\neg P$): La velocidad de conexión a internet **no** es de al menos 25 Mbps.

Análisis de inferencia ejercicio 8, Modus Tollendo Tollens (MTT): Si Thor, Hulk, Iron Man y Capitán América forman equipo, entonces el mundo está a salvo. El mundo no está a salvo.

1. **Variables proposicionales:**

P: Thor, Hulk, Iron Man y Capitán América forman equipo.

Q: El mundo está a salvo.

2. **Regla de inferencia:**

Forma	Nombre
$(P \implies Q) \wedge \neg Q$	Modus Tollendo Tollens (MTT)
$\therefore \neg P$	

3. **Simbolización y derivación:**

- a. **P1** : $P \implies Q$ (Si los 4 héroes forman equipo, el mundo está a salvo)
- b. **P2** : $\neg Q$ (El mundo **no** está a salvo)
- c. **C** : $\neg P$ (**Por MTT** en a y b)

Conclusión ($\neg P$): Thor, Hulk, Iron Man y Capitán América **no** formaron equipo.

Finalización del capítulo 1