**Lógica proposicional y álgebra booleana:**

Según Jose Labra G. y Ana I. Fernández (1998) "la lógica Proposicional pretende estudiar las frases

declarativas simples que son los elementos básicos de transmisión de conocimiento humano" (p. 3)

En conclusión, las frases declarativas deben ser falsas o verdaderas, importante en álgebra booleana.

Otro conocimiento importante de la lógica proposicional en álgebra booleana son las tablas de verdad, la cuales son descritas en el acta del XVIII Simposio Internacional de informática educativa por

Susana Nieto e Higinio Ramos (2016):

"Las tablas de verdad son una herramienta básica en el análisis de las proposiciones compuestas, dentro del campo de la lógica proposicional. Pero su utilidad se extiende más allá, abarcando aspectos de la lógica de Boole, la teoría de circuitos, ..." (p. 97)

Como un tercer punto, las operaciones que se pueden realizar con las proposiciones corresponden al álgebra booleana ya que las conjunciones, disyunciones y negaciones, se traducen directamente de las palabras a simbología matemática, así que "y" es traducido como operación conjunción, la "o" es interpretado como disyunción y "no" se traduce como negación. (Bradford, Henry Arnold, 2011)

**Aplicaciones en informática:**

* El álgebra booleana en informática es utilizada para la toma de decisiones de un programa o código, utilizando condiciones las cuales ejecutan parte de un programa en caso de que la condición sea verdadera.
* El álgebra booleana también es importante en base de datos ya que los registros de una consulta se pueden filtrar utilizando álgebra booleana, realizando una combinación de operaciones lógicas básicas AND, OR o NOT.
* Otra de las aplicaciones del álgebra booleana en informática es utilizado en el mundo de la criptografía, ya que cada vez más personas utilizan el internet, se hace necesario que los canales de comunicación sean seguras (Crama, Ives y Hammer, Peter,2010)

**Comentario 1:**

¡Buenas Joshua!

Profundizando lo que mencionas, aporto lo siguiente, desde mi experiencia como programador:

La lógica proposicional es necesario en Ingeniería en Informática, ya que:

1. Muchas veces se define dentro de un programa cuales bloques de código se ejecutan y cuáles no. Un ejemplo de proposición es la siguiente, si se desea que un bloque de código corra en caso de que: Un número sea mayor a diez "y" a su vez que sea un número positivo (esto corresponde a una proposición compuesta). En este caso vemos que su tabla de verdad corresponde a una conjunción ya que se utiliza la palabra "y", por lo que para que la proposición sea verdadera, se necesita lo siguiente:  
   - Un número sea mayor a diez, que se puede denominar p  
   - Debe ser un número positivo, que se puede denominar q

Solamente en caso de que ambos p y q sean verdaderos, el bloque de código se ejecuta, en caso contrario el bloque de código es ignorado.  
La operación lógica es entonces: p ^ q, la cual representa una operación "Y" lógica (en inglés AND).

1. Se puede realizar operaciones dentro de un programa en un bucle "for", ese bloque de código se repite n veces hasta que una condición sea alcanzada. Por ejemplo, la proposición: Repetir bloque de código hasta que x sea igual a 0, esto es útil para realizar búsquedas y el ciclo se detendrá hasta que la condición sea verdadera.  
   Lo último mencionado va muy de la mano con lo que mencionas referente a la creación de ciclos.

**Comentario 2:**

La lógica proposicional es necesario en Ingeniería en Informática, ya que:

1. Por ejemplo, si un programa que se ejecuta posee una variable que no está definida o sea indeterminada, que sea posible salir del programa, e indicar mediante un mensaje de error indicando por qué se dio la terminación del programa (en este caso sería debido a que el valor de una variable no se ha definido). La proposición en este caso sería: La variable x se encuentra definida. Aquí vemos como las opciones para la proposición anterior es Verdadero o Falso. En caso de ser Falso, esto indica que la variable x no esta definida. Esto es comúnmente utilizado para manejo de excepciones o errores.
2. En el desarrollo de páginas web con acceso mediante un usuario y una contraseña, para usuarios administradores, las opciones dentro de la página serán más variada, por otro lado, si el usuario ingresado es un estudiante, sólo será capaz de observar los cursos matriculados, las notas y las tareas de dicho curso, etc.; es decir para un estudiante la herramienta web tendrá menos información. Esto es posible gracias a las proposiciones lógicas siguientes:

* ¿El usuario que ingresó es administrador? (Las opciones son verdadero o falso)
* ¿El usuario que ingresó es estudiante? (Las opciones son verdadero o falso)