

Acad Myia Pugao Progurmo 1/6

Title Introducción

Keyword Topic 5.1

algebra booleana
Computación
circuitos lógicos
operadores
lógicos,
Claude Shannon

El álgebra booleana fue desarrollado en 1854 por George Boole como una estructura matemática para representar proposiciones lógicas a través de símbolos algebraicos. Aunque en su inicio no tuvo aplicaciones prácticas, en 1938 Claude Shannon demostró que el álgebra de Boole podía aplicarse al diseño de circuitos lógicos y computadoras, sentando las bases para la computación digital moderna.

Questions

¿Qué es el álgebra booleana?
¿Qué aplicaciones tiene en la computación?
¿Cuáles son los operadores básicos del álgebra booleana?

AND
OR +
NOT -

Summary:

El álgebra booleana es clave en el diseño de procesadores, memoria RAM, sistemas de automóviles y cualquier sistema basado en lógica.

NAME CLASS SPEAKER DATE & TIME
alex myia Pigeo. Programacion 2/6

Title Expresiones Booleanas

Keyword Topic S.2

VARIABLES binarias
expresiones
booleanas,
operadores
lógicos,
circuitos
digitales

Las expresiones son formadas por variables binarias que solo pueden tomar valores de 0 (falso) o 1 (verdadero), combinados con los operadores lógicos.

Ejemplo de una expresión booleana:

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

A+B representa una operación OR

C representa la negación de C.

(A + B) · C representa una operación AND entre ambos términos

Questions

¿Cuáles pueden
tomar los
variables booleanos?

¿Cómo se repre-
sentan una opre-
ción AND en una
expresión boolé-
ana?

¿En qué aplicacio-
nes se usan las
expresiones booleanas?

Summary: Estas expresiones se utilizan para modelar el comportamiento de circuitos electrónicos, donde cada variable representa el estado de un interruptor o componente digital.

NAME *alex myo Byoo Programación* CLASS *3/b* SPEAKER DATE & TIME

Title *Propiedades de los Expresiones Booleanas*

Keyword Topic *5.3*

Commutatividad
asociatividad
distributividad
leyes de morgan
simplificación

El álgebra booleana sigue ciertas propiedades fundamentales que permiten simplificar expresiones y optimizar circuitos.

1. Commutatividad:

$$A + A = A ; A \cdot A = A$$

(Un elemento OR o AND consigo mismo no cambia)

Commutatividad: $A + B = B + A ; A \cdot B = B \cdot A$
(el orden de los operadores no afecta el resultado)

Asociatividad: $(A + B) + C = A + (B + C) ; (A \cdot B) \cdot C = A \cdot (B \cdot C)$

Distributividad: $A \cdot (B + C) = A \cdot B + A \cdot C$

Leyes de Morgan: $(A + B)' = A' \cdot B' ; (A \cdot B)' = A' + B'$

(Transforman una operación AND en OR y viceversa al ocurrir una negación)

Summary: Estas propiedades se utilizan para la simplificación de circuitos digitales y para diseñar sistemas más eficientes.

NAME *Axel Munguia Pugro* CLASS *Programación* SPEAKER *1/b* DATE & TIME

Title *Optimización de expresiones Booleanas*

Keyword Topic 5.4
optimización, simplificación, mapos de Karnaugh, circuitos digitales y expresiones Booleanas

Para minimizar circuitos digitales y expresiones Booleanas, se emplean dos métodos principales:

- 1. Simplificación Algebraica: Se utilizan las propiedades del álgebra Booleana para reducir el número de términos en una expresión.*

$$\text{Ej: } AB + A'B + AB' = A + B$$

2. Mapos de Karnaugh:

Son tablas gráficas que permiten visualizar y agrupar términos con valores iguales, facilitando la simplificación sin necesidad de manipulación algebraica extensa.

Questions

¿Qué técnicas se usan para optimizar expresiones Booleanas?

¿Cómo funcionan los mapos de Karnaugh?

<i>f</i>	<i>AB</i>	<i>00</i>	<i>01</i>	<i>11</i>	<i>10</i>
		0	1	1	0

Summary: *Esto reduce el número de compuertas lógicas necesarias en un circuito.*

NAME alex mijis Pugno Programación CLASS 5 SPEAKER S/p DATE & TIME

Title Computadores lógicos

Keyword Computadores Topic S.5

Computadores

lógicos,

Circuitos digitales

AND, OR, NOT,

XOR.

AND:

Símbolo: 

Tabla de verdad:

A	B	$A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

$$f = A \cdot B$$

OR

$$f = A + B$$

Símbolo: 

Tabla de verdad:

A	B	$A + B$
0	0	0
0	1	1
1	0	1
1	1	1

NOT:

$$P = A'$$

Símbolo: 

Tabla de verdad:

A	A'
0	1
1	0

Questions

Que función
realiza cada
Computador lógico?
¿Cuál es la dife-
rencia entre
AND y NAND?

Summary: Existe también computador combinados como Nand, Nor, Xor y Xnor, mencionados en el libro de microprocesadores y memoria Ram.

NAME
alcel myiafigo
CLASS
6/6
SPEAKER
DATE & TIME

Title Apliaciones del álgebra Booleana

Keyword Topic S.6

El álgebra booleano es la base del diseño de hardware digital y tiene aplicaciones en:

1. Computadoras

Procesadores usan lógica booleana para ejecutar operaciones - circuito de memoria almacenar información en estados 0 y 1.

2. Sistema de control

Semáforos y ascensores funcionan con reglas booleanas para activar o desactivar señales.

Questions

¿Cómo se usa el álgebra booleana en los computadores?

3. Electrónica Digital.

Los circuitos lógicos se utilizan en calculadoras, relojes digitales, microcontroladores y sistemas imbibidos

¿Qué relación tiene con los sistemas de control industriales?

4. Automatización Industrial

Se utiliza en PLC

Summary: Se usan en computadoras, sistemas de control automatización y electrónica digital. Permiten diseños procedentes, memoria y sistemas de seguridad.

NAME _____ CLASS _____ SPEAKER _____ DATE & TIME _____
acel Mayas Rayos Progiomorfos 1/b

Title

Introducción

Keyword Topic 6.1

Sistema numérico

histórico,
decimal,
binario

Computación

Los sistemas de numeración han evolucionado
alo largo de la historia. Los primeros pueblos
usaron simblos como rayos, círculos y
figuras para representar cantidades.

Con el tiempo, se adoptaron sistemas
más organizados como:

Sistema vigesimal (basado en simblos para diferentes valores: I = 1, X = 10, Z = 100).

Questions

¿Cuáles son los
sistemas de nu-
meración más
antiguos?

¿Qué diferencia
hay entre un
sistema posicional
y no posicional?

Sistema romano (I, V, X, L, C, D, M)

Sistema babilónico (Sexagesimal, base 60)

Sistema maya (Sexagesimal, base 20)

Summary:

Estos sistemas antiguos tienen problemas al
manejar números grandes ya que se desarrollaron
sistemas posicionales, donde el valor de un número
depende de su posición en la cifra.

NAME: alexis miguel Ríos CLASS: Programación SPEAKER: 2/6 DATE & TIME:

Title: Sistemas Numéricos Posicionales

Keyword: Base numérica Topic: 6.2

Base decimal,
binaria,
octal,
hexadecimal.

Los sistemas posicionales usan una base específica y un conjunto de simbolos. La expresión general de un numero en un sistema:

$$N = d_n b^n + d_{n-1} b^{n-1} + \dots + d_1 b^1 + d_0 b^0$$

Ej de sistemas posicionales

Decimal (base 10) usa los dígitos 0-9.

Questions

¿Como funcionan
en sistema
numeroso posicional (base 8): usa 0-7
decimal?

¿Qué bases se
usen en com-
putación?

Binario (base 2) usa 0 y 1.

Octal (base 8): usa 0-7

Hexadecimal (base 16): usa 0-9 y A-F (A=10, B=11, ..., F=15)

Summary: Estos sistemas son fundamentales en computación y electrónica, ya que los computadores trabajan con sistemas binarios, y los sistemas octal y hexadecimal facilitan la conversión y manipulación de datos.

NAME

CLASS

SPEAKER

DATE & TIME

alexel myo figo (Programación) 3/b

Title Conversion entre sistemas de numeración

Keyword

Topic 6.3

Conversion

decimal a binario

binario a decimal

decimal, octal, hexadecimal

Se pueden convertir numeros entre diferentes sistemas siguiendo métodos específicos:

conversion de decimal a otra base

- se divide el numero entre la base destino

y se obtiene los residuos.

- se lee el resultado en orden inverso.

ej: convertir 45_{10} a binario:

$$45/2 = 22 \text{ residuo } 1$$

$$22/2 = 11 \text{ residuo } 0$$

$$11/2 = 5 \text{ residuo } 1$$

$$5/2 = 2 \text{ residuo } 1$$

$$2/2 = 1 \text{ residuo } 0$$

$$1/2 = 0 \text{ residuo } 1$$

Questions

¿Cómo convertir
un numero decimal a binario?

7. conversión de una base a decimal

¿Qué método se
usa para convertir

$$1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$$

$$= 32 + 0 + 8 + 4 + 1 = 45_{10}$$

binario a octal?

3. conversión entre bases que son potencias de 2

$$Ej: 101101_2$$

$$\text{equivalente: } 101101_2$$

$$\text{Convertimos: } 101_2 = 5_8, 101_2 = 5_8$$

$$\text{Resultado: } 558$$

Summary: Se pueden convertir numeros entre bases usando división sucesiva, potencias y agrupación de bits (as bases 2, 8 y 16).

NAME: *arcelmojor Pérez* CLASS: *Promoción* SPEAKER: *4/6* DATE & TIME:

Title: *Operaciones en diferentes Bases*

Keyword Topic 6.4

Suma binaria,
resta,
multiplicación,
división,
números de conversión

Suma en Binario
Se aplican los siguientes reglos:
 $0+0=0$ $0+1=1$ $1+1=10$ (lleva 1)
 $1+1+1=11$ (lleva 1)

Ej: $1101_2 + 1011_2$

$$\begin{array}{r} 1101 \\ + 1011 \\ \hline 11000 \end{array}$$

Resultado: 11000_2 (equivalente en decimal)

Resta en Binario

Se usa la técnica de prestamo

Questions

¿Cómo se realiza una suma binaria?

multiplicación en Binario
Se usa el mismo método que en decimal,
pero solo con 0s y 1s

¿Cómo se maneja la división en sistemas numéricos distintos?

División en Binario
Se usa el mismo algoritmo de división que en decimal, restando potencias de 2

Summary:

Las operaciones aritméticas en diferentes bases siguen los mismos reglos que en decimal, pero con modificaciones según la base.

NAME CLASS SPEAKER DATE & TIME
Axel Mejia Pugoro Programación 5/6

Title Complemento a 2 y Representación de números negativos

Keyword Topic 6.5

Complemento a 2, los números negativos, En computación, los números negativos se representan con el complemento a 2:

1. Tomar el complemento a 1 (complemento 0 para y viceversa).
2. Sumar 1 al resultado.

Ej: Representar 5_{10} en 8bits

1. escribir su binario 100000101_2

2. complemento a 1: 11111010_2

3. Sumar 1: 11111011_2

Resultado: $5_{10} = 11111011_2$

Questions

Porque se usa el complemento a 2 en computación?

Como convertir un numero a complemento a 2?

Summary: El complemento a 2 permite representar números negativos en computadoras, conservando valores positivos en su representación negativa.

NAME CLASS SPEAKER DATE & TIME
Axel Miyazawa Programación b/b
Title - Aplicaciones de los sistemas de numeración

Keyword Topic b.b

Computación,

nodos,

programación,

cryptografía,

datos.

Los sistemas de numeración tienen

aplicaciones en:

1. Computación y programación

- Binario: lenguaje binario de los computadores

- Octal y hexadecimales. Representación

compuestos en ensamblador y dirección

de memoria

2. Electrónica digital

- uso en microprocesadores, registros y

operadores lógicos

3. Códigos y comunicaciones

- Códigos ASCII y univocados usados

sistemas numéricos para representar

caracteres

Questions

¿Cómo se usan
los sistemas no-
mericos en progra-
mación?

¿Cómo se aplican
en la transmisión
de datos?

1. Cryptografía y seguridad informática

- Algoritmo de cifrado como AES usan

binaria binaria y hexadecimal.

Summary: Se usan en computadoras, criptografía, programación
electrónica y comunicaciones digitales.

NAME CLASS SPEAKER DATE & TIME
Axel Mijico Programación 1/9

Title Introducción

Keyword Topic 7.1

Código binario,
Representación
digital,
Computadoras,
Sistemas digitales

Los computadores utilizan el sistema binario
porque los circuitos electrónicos están
diseados para manejar solo dos estados
(estados).

- 0: No hay corriente eléctrica (bajo voltaje)
- 1: Hay corriente eléctrica (alto voltaje)

Ej:

La letra A en ASCII se representa 01000001
en número decimal como 53. El con-

¿Por qué el sistema
binario es fundamental en computo-
ción?

Los sistemas electrónicos digitales como
memoria, procesadores y uds
dependen completamente del código
binario.

Questions

El código binario es la base de la computación
moderna. Todo lo que usan los computadores se representa
se representa con ceros (0) y unos (1), ya que los circuitos
electrónicos solo pueden detectar dos estados.

NAME

CLASS

SPEAKER

DATE & TIME

Alicel Mejia P. yoro Puigromero

3/9

Title Conversion de numeros a código binario

Keyword**Topic** 7.2

Conversion binaria, séptima decimal, división sucesiva, multiplicación por 2.

método de conversión:

1. División sucesiva (para enteros):

- Se divide el numero entre 2 y se anotan los residuos.

- Se leen los residuos en orden inverso.

Ej: convertir 45₁₀ a binaria.

$$45/2 = 22, \text{ resultado } 1 \quad 22/2 = 11, \text{ residuo } 0$$

$$11/2 = 5, \text{ resultado } 1 \quad 5/2 = 2, \text{ residuo } 1$$

$$2/2 = 1, \text{ resultado } 0 \quad 1/2 = 0, \text{ residuo, 1}$$

Resultado: 45₁₀ = 10110₂**Questions**

¿Cómo convertir un numero decimal a binario?

2 multiplicación por 2 (para fracciones):

- Si multiplicó la parte decimal por 2

- Si tomo lo entero y se repite el proceso

¿Cuál es el método ero con la parte fraccionaria

Ej: convertir 0.375₁₀ a binario:

$$0.375 \times 2 = 0.75 \text{ (entero } 0\text{)}$$

$$0.75 \times 2 = 1.50 \text{ (entero } 1\text{)}$$

$$0.50 \times 2 = 1 \text{ (entero } 1\text{)}$$

resultado: 0.375₁₀ = 0.011₂**Summary:**

Los números en base decimal se convierten a binario mediante el método de divisiones sucesivas (para enteros) y multiplicación por 2 (para fracciones)

NAME	CLASS	SPEAKER	DATE & TIME
Acel Nyu & yoro Programario		3/9	
Title	Códigos Binarios en Computación		

Keyword	Topic	7.3
ASCII, Unicode, BCD, Código Gray	códigos binarios más utilizados:	
Representaciones binarias	1. ASCII	- Representa caracteres con 8 bits - Ej: "A" = 0100 0001, "B" = 0100 0010
	2. Unicode	- Expande ASCII para soportar múltiples alfabetos
	3. BCD (Binary Coded Decimal)	- Representa números decimales en binario - Ej: 237 en BCD es 0010 0011.
	4. Código Gray	- Reduce errores en sensores digitales. - Ej: Decimal 0 → Gray 0000 Decimal 1 → Gray 0001 Decimal 2 → Gray 0011 Decimal 3 → Gray 0110
Questions		
¿Qué tipos de códigos binarios existen?		
¿Cómo se representa un carácter en ASCII?		

Summary: Los códigos binarios permiten representar texto, números y otros en computadoras. Algunos de los más usados son ASCII, Unicode, BCD y Código Gray.

NAME CLASS SPEAKER DATE & TIME
Alvaro Mejia Pizarro Programacion 21/9

Title Aplicaciones del código binario

Keyword Topic 7.4

Computadoras
redes, almacenamiento digital
teleinformatica,
logica digital

aplicaciones principales

1. Computadoras y software
 - Todos los programas se almacenan y ejecutan en binario.
 - El un archivo de texto usa códigos binarios ASCII.

2. Redes y telecomunicaciones
 - La información se transmite como señales binarias (modulación digital).
3. Almacenamiento digital
 - Discos duros, memoria USB y SSD almacenan información en bytes.

Questions

¿Como se usa el código binario en el almacenamiento digital?

1. Sistema de control
 - Se usa en PLC para la automatización de fábricas y control de tráfico.

¿Porque es importante en telecomunicaciones?

Summary: El código se usa en computadoras, telecomunicaciones, almacenamiento digital y sistema de control.

NAME CLASS SPEAKER DATE & TIME
Axel Moya Ayrosa Programación 3/9

Title Representación de caracteres

Keyword Topic 7.5

Representación de caracteres

ASCII

unicodi

EBCDIC

codificación

Los caracteres (letras, números y símbolos) deben representarse en código binario para ser procesados. Algunos códigos usados incluyen:

1. ASCII (American Standard Code for Information Interchange)

- Usa 7 o 8 bits por carácter.

- Ej: "A" = 0100001, "B" = 01000010

Questions

¿Qué es un código de caracteres?

¿Cuáles son los principales estándares de representación de caracteres?

3. EBCDIC

- Usado en máquinas de IBM, similar a ASCII pero con diferentes asignaciones

2. Unicode

- Expande ASCII para soportar caracteres de múltiples idiomas

- Ej: "ñ" en Unicode es 11110000 10111111 10011011 10000000

Summary:

Los caracteres en computación se representan usando códigos binarios standarizado como ASCII, Unicode y EBCDIC, permitiendo la compatibilidad entre sistemas.

NAME	CLASS	SPEAKER	DATE & TIME
Axel Mijia Ojorges	Pragocomision	6/9	

Title: Representación de números

Keyword	Topic 7.6
Representación numérica binaria	Existen diferentes métodos para representar números en computación.
complemento	1. Entero signo
a) punto flotante	- Representado en binario puro. - Ej: 5,0 = 1011 2. Entero con signo y complemento a 2 - Representa números negativos de forma científica. Ej: 50 en 8 bits \rightarrow 1111011
Questions	3. Punto flotante (IEEE 754) - Usado para representar números del todo. - Ej: 3,14 en IEEE 754 01000000100000000000000000000000 01110110000000000000000000000000
¿Cómo se representa los números enteros en binaria?	Este método permite calcular más rápidos.
¿Qué es el complemento a 2 y para qué sirve?	9. Fuentes de errores digitales.

Summary: Los números en computación se representan en diferentes formatos, como binario puro, complemento a 2 y punto flotante.

NAME	CLASS	SPEAKER	DATE & TIME
Acilayna Figueroa	Pepperwood	7/9	

Title Operaciones con Código Binario

Keyword	Topic
Operaciones binarias	Suma binaria Resta binaria Multiplicación División
	$0+0=0$ $0+1=1$ $1+1=10$ Resta binaria Se suma el complemento de uno resta Ejemplo: $101_2 - 100_2 = 1_2$ - $101_2 = 010_2$, $1_2 = 001_2$ - Complemento de 100_2 es 111_2
	Multiplicación binaria - Similar a la multiplicación decimal - Se suman los dígitos multiplicados - Ej: $110_2 \times 100_2 = 110_2$ (equivalente a $3 \times 2 = 6$)
Questions	División binaria - Similar a la división larga en decimal - Ej: $100_2 \div 10_2 = 10_2$ (equivalente a $12 \div 2 = 6$)

Summary: Los computadores realizan operaciones matemáticas incluyen suma, resta, multiplicación y división, aplicando reglas específicas.

NAME	CLASS	SPEAKER	DATE & TIME
Axel Moya Aguirre	Programación	8/9	

Title Encuentro en la Transmisor de Datos

Keyword	Topic
funciones de transmisión.	Los datos en la transmisión de datos pueden ser una señal - radiofotomagnética - interfaz entre el teléfono y el ordenador.
delicación de errores.	- código de Hamming - para detectar errores y corregirlos.
corrección de errores.	Comunicación
Código Hamming	para detectar y corregir errores se usan los siguientes métodos:
Paridad	1. Paridad de paridad
	- se añade un bit extra para hacer que el número total de 1 sea par (paridad par) o impar (paridad impar).
Questions	Paridad par: 1010110
¿Qué tipos de errores pueden ocurrir en la transmisión de datos?	Paridad impar: 1010111
¿Cómo funciona la corrección de errores?	Paridad de Hamming
	- para el ordenador se usa el código de Hamming
	bit de redundancia
	El número binario 11011 - código paridad 1011001
	11011001
	usar divisiones polinómicas para detectar errores en datos largos
	o usar un auto-correlacionamiento digital.

Summary: Describir la formación de datos, señalar los errores que ocurren al transmitir datos, interfaz entre el ordenador y el teléfono y corrección de errores para garantizar la fiabilidad de los datos.