Matemáticas Discretas

Introducción a las matemáticas discretas

Definición de Matemáticas Discretas

Las Matemáticas Discretas constituyen un área fundamental en la teoría matemática que se centra en objetos y estructuras discretas, en contraste con las matemáticas continuas que se ocupan de conceptos como el cálculo.

En este contexto, nos enfocamos en conjuntos **finitos** o **contables** de elementos.

Importancia de Matemáticas Discretas en Big Data

Optimización de Procesos: Las técnicas discretas, como la teoría de grafos, son esenciales para optimizar procesos en Big Data, ya que permiten representar relaciones y conexiones de manera estructurada.

Seguridad de Datos: La criptografía, una rama de las Matemáticas Discretas, desempeña un papel crucial en la seguridad de datos al garantizar la confidencialidad y la integridad de la información en entornos de Big Data.

Conceptos Fundamentales

- Lógica y Álgebra Booleana
- Teoría de Conjuntos
- Relaciones y Funciones
- Combinatoria y Principio de Inclusión-Exclusión
- Teoría de Grafos
- Teoría de Números
- Probabilidad y Estadística

Lógica y Álgebra Booleana

Introducción

Los circuitos electrónicos se basan en fenómenos físicos relacionados con la presencia y el flujo de la carga eléctrica.

Estos mecanismos son la base de los modernos ordenadores digitales. Conceptos como tensión y la corriente se utilizan para referirse a las propiedades cuantificables de la electricidad.

Transistor

El mecanismo utilizado para controlar el flujo de corriente eléctrica es un dispositivo semiconductor conocido como **transistor**.

Los **transistores** utilizados en los circuitos digitales funcionan como un interruptor de encendido y apagado que se acciona electrónicamente en lugar de mecánicamente. Es decir, a diferencia de un interruptor mecánico que se abre y se cierra en función de la fuerza mecánica aplicada, un transistor se abre y se cierra en función de la **tensión** aplicada.

Circuitos digitales

¿Cómo se construyen los **circuitos digitales**? Un transistor tiene dos estados posibles: la corriente fluye o no fluye. Por lo tanto, los circuitos se diseñan utilizando un sistema matemático de dos valores conocido como álgebra booleana.

Álgebra Booleana

El álgebra booleana es un álgebra para la manipulación de objetos que pueden tomar sólo dos valores, típicamente verdadero y falso, aunque puede ser cualquier par de valores.

Dado que los ordenadores se construyen como colecciones de interruptores que están "encendidos" o "apagados", el álgebra booleana es una forma natural de representar la información digital. Es habitual interpretar el valor digital 0 como falso y el valor digital 1 como verdadero.

Operaciones

Además de los objetos binarios, el álgebra booleana también tiene operaciones que se pueden realizar sobre estos objetos, o variables. Combinando las variables y los operadores se obtienen **expresiones booleanas.**

Una expresión booleana normalmente tiene uno o más valores de entrada y produce un resultado. Tres operadores booleanos comunes son **AND**, **OR** y **NOT**.

Operadores: Tabla de la verdad

Un operador booleano puede describirse completamente mediante una tabla que enumera las entradas, todos los valores posibles para estas entradas y los valores resultantes de la operación para todas las combinaciones posibles de estas entradas. Esta tabla se llama **tabla de verdad.**

Una tabla de verdad muestra la relación, en forma tabular, entre los valores de entrada y el resultado de un operador o función booleana específica sobre las variables de entrada.

Operadores: Tabla de la verdad

AND Truth Table

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

OR Truth Table

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

XOR Truth Table

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

NOT Truth Table

A	B
0	1
1	0

Propiedades

Aparte de lo anterior dicho, por ser álgebra de Boole (A) se cumplen las siguientes propiedades, por simplificación la operación OR es + y la operación AND es x.

- **Conmutativa**: Para todo a,b perteneciente a A, $a+b=b+a$ y $a \times b=b \times a$.
- **Identidad**: Para todo a pertenece A, $a+0=a$ y $a \times 1=a$.
- **Distributiva**: Para todo a,b,c pertenece A, $a+(b\times c)= (a+b)\times a+c)$ y $a\times(b+c)= (a\times b)+(a\times c)$
- **Complementaria**: Para todo a que pertenece A, $a+a= 1$ y a×compl.de $a=0$.

Puertas lógicas

Los operadores lógicos AND, OR y NOT de los que hemos hablado se han representado hasta ahora en sentido abstracto utilizando tablas de verdad y expresiones booleanas.

Los componentes físicos reales, o circuitos digitales, se construyen a partir de una serie de elementos primitivos llamados **puertas lógicas**. Las puertas implementan cada una de las funciones lógicas básicas de las que hemos hablado. Estas puertas son los bloques de construcción básicos para el diseño digital.

Puertas lógicas

Una puerta implementa un operador booleano simple. Para implementar físicamente cada puerta se necesitan de uno o varios transistores dependiendo de la tecnología que se utilice.

Símbolos de las puertas lógicas

En primer lugar, examinaremos las tres puertas más sencillas. Éstas corresponden a los operadores lógicos AND, OR y NOT. Hemos discutido el comportamiento funcional de cada uno de estos operadores booleanos.

Otras dos puertas comunes son NAND y NOR, que producen una salida complementaria a AND y OR, respectivamente. Cada puerta tiene dos símbolos lógicos diferentes que pueden utilizarse para la representación de la puerta.

Símbolos de las puertas lógicas

Otra puerta común es la puerta de OR exclusivo (XOR), representada por la expresión booleana $x \oplus y$. XOR es falsa si ambos valores de entrada son iguales y verdadero en caso contrario.

Símbolos de las puertas lógicas

Name	NOT	AND	NAND	OR	NOR	XOR
Alg. Expr.	$\overline{A}$	AB	$\overline{AB}$	$A+B$	$\overline{A+B}$	$A \oplus B$
Symbol						
Truth Table	A X 0 1 1 0	B A X 0 0 0 0 1 0 1 0 0 1 1 1	B A X 0 0 1 0 1 1 1 0 1 1 1 0	B A X 0 0 0 0 1 1 1 0 1 1 1 1	B A X 0 0 1 0 1 0 1 0 0 1 1 0	B A X 0 0 0 0 1 1 1 0 1 1 1 0

Símbolos de las puertas lógicas

En nuestros ejemplos hasta ahora, todas las puertas han aceptado sólo dos entradas. Sin embargo, las puertas no se limitan a dos valores de entrada. Hay muchas variaciones en el número y los tipos de entradas y salidas permitidas para varias puertas

Representar información

Los sistemas de numeración representan valores numéricos.

Existen dos tipos: posiciones y no posicionales.

ROMANO	DECIMAL
I	1
V	5
X	10
L	50

Representar información

Sistema **decimal**: 121. Los dígitos 1 no representan el mismo valor. 121= 1*100 + 2*10 + 1

Sistema **romano**: CXXI = 100 + 10 + 10 +1 =121. En este caso todos los dígitos X representan la misma cantidad, independientemente de su posición.

Sistema decimal

El sistema de numeración decimal está compuesto por 10 símbolos, estos son 0, 1, 2, 3, 4, 5, 6, 7, 8 y 9.

1456_{10}

Sistema binario

El sistema binario es un sistema de numeración que tan solo utiliza dos dígitos: 0 y 1.

Es el sistema de numeración usado por el ordenador, de base 2.

Convertir a base decimal: 1001_2

Sistema Octal

El sistema octal es un sistema de numeración que tan solo utiliza 8 dígitos: 0, 1, 2, 3, 4, 5, 6, 7 y 8.

Convertir a base decimal: 772_8

Sistema Hexadecimal

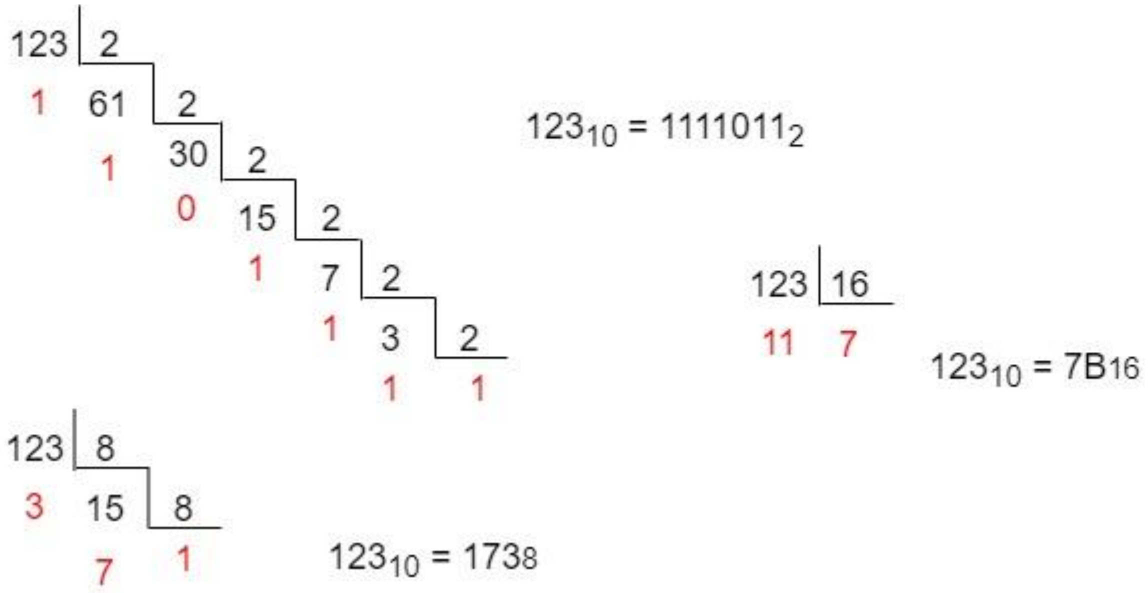
El sistema hexadecimal es un sistema de numeración que utiliza 16 dígitos: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E y F.

Convertir a base decimal: $12F_{16}$

Conversión de números decimales a binarios, octales y hexadecimales

Para convertir un valor decimal a cualquiera de las bases estudiadas se deben realizar divisiones enteras sucesivas del número decimal entre la base a convertir.

Conversión de números decimales a binarios, octales y hexadecimales



Conversión de números decimales a binarios, octales y hexadecimales

Ejercicio 1) Haz una tabla que relacione los 20 primeros números de los sistemas de numeración decimal, binario, octal y hexadecimal.

Ejercicio 2) Convertir el número 749_{10} a binario, octal y hexadecimal.

Conversión de números decimales a binarios, octales y hexadecimales

Pero, ¿qué pasa si el número tiene decimales?

Ejemplo: convertir a binario $123{,}3125_{10}$

Aplicando el método anterior tenemos que $123=1111011_2$

Entonces para calcular la parte decimal haremos el siguiente método

0,3125*2 =	0,	625
0,625*2 =	1,	25
0,25*2 =	0,	5
0,5*2 =	1,	0

$123{,}3125_{10} = 1111011{,}0101_2$

Conversión de números decimales a binarios, octales y hexadecimales

Ejercicio 3) Convertir el número $745,27_{10}$ a binario, octal y hexadecimal.

Ejercicio 4) Convertir el número $129,567_{10}$ a binario, octal y hexadecimal.

Conversión de binario a octal y hexadecimal

Podemos convertir un número binario a octal o hexadecimal agrupando 3 o 4 bits respectivamente.

10011_2 a octal

El número binario se separa en grupos de 3 bits

10011_2=010|011

A continuación cada binario de tres dígitos se convierta a decimal

010=2 y 011=3

Por lo tanto tenemos que $10011_2=23_8$

Conversión de binario a octal y hexadecimal

Podemos convertir un número binario a octal o hexadecimal agrupando 3 o 4 bits respectivamente.

10011_2 a hexadecimal

El número binario se separa en grupos de 4 bits

10011_2=0001|0011

A continuación cada binario de cuatro dígitos se convierta a decimal

0001=1 y 0011=3

Por lo tanto tenemos que $10011_2=13_{16}$

Ejercicios

Ejercicio 5) Convertir 1011011001101.00101011_2 a hexadecimal

Ejercicio 6) Convertir $27CB,0A_{16}$ a binario.

Ejercicio 7) Convertir $373,6_8$ a hexadecimal.

Ejercicio 8) Convertir $6A,D_{16}$ a octal.

Ejercicio 9) Convertir el número $11011001001,1101_2$ a hexadecimal.

Ejercicios

Ejercicio 10) Convierte $10111{,}01_2$ a decimal.

Ejercicio 11) Convierte $52{,}375_2$ a binario.

Ejercicio 12) Convierte $654{,}40625_8$ a decimal.

Ejercicio 13) Convierte $123{,}45_{10}$ a octal.

Ejercicio 14) Convierte $1B3{,}2_{16}$ a decimal.

Ejercicio 15) Convierte $251{,}625_{10}$ a hexadecimal.

Relaciones y Funciones

Definición de Relaciones

Una relación en matemáticas discretas es un conjunto de pares ordenados, donde cada par ordenado establece una conexión entre dos elementos de conjuntos diferentes. Formalmente, dada una relación R entre dos conjuntos A y B, se representa como $R \subseteq A \times B$, donde $A \times B$ es el conjunto de todos los pares ordenados (a, b) con $a \in A$ y $b \in B$.

Ejemplo de Relación

Considera los conjuntos $A = \{1, 2, 3\}$ y $B = \{a, b, c\}$. Definimos la relación R como el conjunto de pares ordenados $\{(1, a), (2, b), (3, c)\}$. En este caso, cada elemento en el conjunto A está relacionado con un elemento correspondiente en el conjunto B. Esta relación puede representarse como $R \subseteq A \times B$.

Definición de función

Una función es un tipo especial de relación en la que cada elemento del conjunto de entrada (dominio) se asigna a exactamente un elemento en el conjunto de salida (contradominio).

Formalmente, si tenemos una función $f:A \rightarrow B$, cada elemento a en A tiene una única imagen b en B.

Ejemplo de función

Tomemos la función $f:R \rightarrow R$ definida por $f(x)=2x+1$. En esta función, cada número real x tiene una única imagen en R dada por $2x+1$.

Por ejemplo, si tomamos $x=3$, entonces $f(3)=2\times3+1=7$. No importa cuántas veces seleccionemos el mismo valor de x, la función asignará siempre el mismo valor.

. Propiedades de Relaciones

- **Reflexividad**: Una relación R es reflexiva si cada elemento de A está relacionado consigo mismo. Formalmente, $\forall a \in A$: $(a,a) \in R$.
- **Simetría**: Una relación R es simétrica si, para cada par ordenado (a, b) en R, también está (b, a). Es decir, $\forall (a,b) \in R$, $(b,a) \in R$.
- **Transitividad**: La relación R es transitiva si, siempre que (a, b) y (b, c) están en R, entonces (a, c) también está en R. Esto se expresa como $\forall (a,b) \in R, (b,c) \in R \Rightarrow (a,c) \in R$.

Ejemplo de relación

Dado un conjunto A (por ejemplo, el conjunto de números reales), consideramos la relación de igualdad R en A. Esta relación se define como el conjunto de pares ordenados (a,b) donde a=b.

Formalmente, $R=\{(a,b) \mid a,b \in A \text{ y } a=b\}$

¿Se cumplen las propiedades?

Funciones y sus Propiedades

Una función f de A en B, denotada como $f:A \rightarrow B$, asigna a cada elemento de A exactamente un elemento de B.

- **Dominio**: El conjunto de todos los elementos de A que tienen una imagen bajo la función f.
- **Contradominio**: El conjunto B, que contiene todas las posibles imágenes de los elementos de A bajo la función f.
- **Imagen**: El conjunto de todas las imágenes posibles de los elementos de A bajo la función f.

Funciones y sus Propiedades

Sea la función $h:\{1,2,3\}\rightarrow\{a,b,c,d\}$ expresada

$f(1)=a$

$f(2)=b$

$f(3)=c$

Calculad, dominio, imagen y contradominio.

Composición de funciones

La composición de dos funciones, f y g, se denota como $(f \circ g)(x)$ y se define como $(f \circ g)(x)=f(g(x))$. Es decir, primero aplicamos la función g y luego aplicamos la función f al resultado.

Ejemplo de composición de funciones

Considera las funciones $f: R \rightarrow R$ y $g: R \rightarrow R$ definidas como:

$f(x)=2x+1$

$g(x)=x^2$

Calcular la composición de estas funciones, denotada como $(f \circ g)(x)$ y $(g \circ f)(x)$.

Funciones Inyectivas, Sobreyectivas y Beyectivas:

- **Inyectiva**: Una función es inyectiva si elementos distintos en el dominio tienen imágenes distintas en el contradominio. Matemáticamente, $f(x_1)=f(x_2)\Rightarrow x_1=x_2$.
- **Sobreyectiva**: Una función es sobreyectiva si cada elemento del contradominio tiene al menos un elemento en el dominio que se le asigna. En otras palabras, la función abarca todo su contradominio.
- **Biyectiva**: Una función es biyectiva si es tanto inyectiva como sobreyectiva, lo que significa que asigna cada elemento único del dominio a un único elemento del contradominio y cubre todo el contradominio.

FIN