04 - Fundamentos de Álgebra - Sistemas de Numeración, Números Primos y Aritmética Modular

Este tema abarca tres conceptos fundamentales en álgebra que están estrechamente relacionados entre sí: los sistemas de numeración en diferentes bases, los números primos como elementos básicos de todos los números naturales, y la aritmética modular como una nueva forma de realizar operaciones matemáticas. Estos conceptos forman la base para entender estructuras algebraicas más complejas y tienen aplicaciones prácticas en informática, criptografía y matemáticas aplicadas. A lo largo de este estudio, desarrollaremos una comprensión profunda de cómo los números pueden representarse de diferentes maneras, cómo se construyen a partir de elementos básicos llamados primos, y cómo podemos trabajar con un tipo especial de aritmética que tiene propiedades únicas.

Sistemas de Numeración y Bases

Concepto Fundamental de Bases de Numeración

Un sistema de numeración es una forma sistemática de representar números utilizando un conjunto limitado de símbolos o dígitos. El sistema que utilizamos cotidianamente es el sistema decimal (base 10), pero existen infinitos sistemas de numeración posibles1. La base de un sistema determina cuántos símbolos diferentes utilizamos para representar los números y cómo interpretamos el valor posicional de cada dígito.

En cualquier sistema de numeración con base b, utilizamos exactamente b símbolos diferentes para representar los números. Por ejemplo, en el sistema decimal **(b=10)** utilizamos los dígitos {**0, 1, 2, 3, 4, 5, 6, 7, 8, 9}**. Cada posición en un número representa una potencia diferente de la base, comenzando desde la potencia 0 en la posición más a la derecha.

La representación matemática de un número en cualquier base sigue el patrón: un número n en base b se expresa como **n = dk·b^k + dk-1·b^(k-1) + ... + d1·b^1 + d0·b^0**, donde cada di es un dígito válido en esa base **(0 ≤ di < b)**. Este sistema posicional permite representar cualquier número natural de forma única utilizando un conjunto finito de símbolos.

El Sistema Decimal como Punto de Partida

El sistema decimal es nuestro sistema de referencia porque lo utilizamos desde la infancia. Cuando escribimos el número 3742, implícitamente estamos expresando: $3742 = 3\times10^3 + 7\times10^2 + 4\times10^1 + 2\times10^0 = 3\times1000 + 7\times100 + 4\times10 + 2\times1 = 3000 + 700 + 40 + 2$. Esta descomposición muestra claramente el valor posicional de cada dígito.

Es fundamental comprender que el valor de cada posición se multiplica por una potencia creciente de la base (10 en este caso) al movernos de derecha a izquierda. La posición más a la derecha corresponde a la **potencia 0**, la siguiente a la **potencia 1**, y así sucesivamente. Esta regla es universal y se aplica a todos los sistemas de numeración.

El Sistema Binario

El sistema binario (**base 2**) utiliza únicamente dos símbolos: **{0, 1}.** Este sistema es fundamental en informática porque los dispositivos digitales pueden representar fácilmente estos dos estados (encendido/apagado, alto/bajo voltaje, etc.). En el sistema binario, cada posición representa una potencia de 2.

Por ejemplo, el número binario 11010_2 se interpreta como: **$11010_2 = 1\times2^4 + 1\times2^3 + 0\times2^2 + 1\times2^1 + 0\times2^0 = 1\times16 + 1\times8 + 0\times4 + 1\times2 + 0\times1 = 16 + 8 + 0 + 2 + 0 = 26$** en decimal. Es importante notar que cada posición vale exactamente el doble que la posición inmediatamente a su derecha.

Para desarrollar intuición sobre el sistema binario, es útil practicar la conversión mental de números pequeños. Los **primeros números binarios** son: **$1_2=1_{10}$, $10_2=2_{10}$, $11_2=3_{10}$, $100_2=4_{10}$, $101_2=5_{10}$, $110_2=6_{10}$, $111_2=7_{10}$, $1000_2=8_{10}$, y así sucesivamente**.

El Sistema Hexadecimal

El sistema hexadecimal (**base 16**) requiere 16 símbolos diferentes. Como solo tenemos 10 dígitos numéricos, utilizamos las **letras A, B, C, D, E, F** para representar los valores 10, 11, 12, 13, 14, 15 respectivamente. Este sistema es

muy utilizado en informática porque cada dígito hexadecimal corresponde exactamente a 4 dígitos binarios.

Un ejemplo ilustrativo es **$CD38_{16} = C\times16^3 + D\times16^2 + 3\times16^1 + 8\times16^0 = 12\times4096 + 13\times256 + 3\times16 + 8\times1 = 49152 + 3328 + 48 + 8 = 52536$** en decimal. La ventaja del hexadecimal es que proporciona una forma compacta de representar números binarios largos, ya que cada dígito hexadecimal equivale a exactamente 4 dígitos binarios.

Conversión Entre Bases

De Cualquier Base a Decimal

Para convertir un número de cualquier base b al sistema decimal, simplemente aplicamos la fórmula de expansión posicional. Cada dígito se multiplica por la potencia correspondiente de la base y se suman todos los productos. Este método es directo pero requiere calcular potencias de la base.

De Decimal a Cualquier Base

Para convertir un número decimal a otra base b, utilizamos el algoritmo de divisiones sucesivas. El proceso consiste en:

1. Dividir el número decimal por la base b, obteniendo un cociente q_o y un resto r_o
2. Dividir el cociente q_o por b, obteniendo q_1 y r_1
3. Continuar este proceso hasta que el cociente sea 0
4. Los restos, leídos en orden inverso, forman el número en la nueva base

Por ejemplo, para convertir 23_{10} a binario: 23÷2=11 resto 1, 11÷2=5 resto 1, 5÷2=2 resto 1, 2÷2=1 resto 0, 1÷2=0 resto 1. Leyendo los restos de abajo hacia arriba: $23_{10} = 10111_2$.

Números Primos y Factorización

Definición y Propiedades Básicas de los Números Primos

Un número primo **es un número natural mayor que 1 que tiene exactamente dos divisores naturales: 1 y él mismo**. Esta definición aparentemente simple

encierra una de las estructuras más fascinantes y complejas de las matemáticas. Los números que no son primos (excepto el 1) se denominan números compuestos, y estos siempre pueden expresarse como producto de dos o más números naturales diferentes de 1 y del propio número.

Los primeros números primos son: 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47.... Es importante notar que 2 es el único número primo par, ya que todos los demás números pares son divisibles por 2 y por tanto no pueden ser primos. El número 1 no se considera primo por convenio matemático, ya que esta exclusión hace que muchos teoremas importantes funcionen de manera más elegante.

El Teorema Fundamental de la Aritmética

Uno de los resultados más importantes en la teoría de números establece que t**odo número natural mayor que 1 puede expresarse de manera única como producto de potencias de números primos**. Formalmente, para cualquier número n > 1, existe una única descomposición $n = p_1^{\alpha_1} \times p_2^{\alpha_2} \times \ldots \times p_k^{\alpha_k}$, donde $p_1 < p_2 < \ldots < p_k$ son números primos y $\alpha_1, \alpha_2, \ldots, \alpha_k$ son exponentes positivos.

Esta descomposición se denomina factorización prima o descomposición factorial del número. Por ejemplo: $12 = 2^2 \times 3^1$, $30 = 2^1 \times 3^1 \times 5^1$, $100 = 2^2 \times 5^2$. La unicidad de esta descomposición significa que no importa el método que utilicemos para factorizar un número, siempre llegaremos a la misma combinación de primos y exponentes.

Algoritmo de Factorización

Para obtener la factorización prima de un número, utilizamos un a**lgoritmo sistemático:**

1. Comenzamos dividiendo el número por el primer primo (2) tantas veces como sea posible
2. Cuando 2 ya no divida al cociente, probamos con el siguiente primo (3)
3. Continuamos este proceso con primos sucesivos hasta que el cociente sea 1

Por ejemplo, para factorizar 126:

- 126 ÷ 2 = 63 (una vez)

- 63 ÷ 3 = 21 (primera vez)
- 21 ÷ 3 = 7 (segunda vez)
- 7 ÷ 7 = 1 (una vez)

Por tanto, $126 = 2^1 \times 3^2 \times 7^1$.

Una forma práctica de organizar este proceso es mediante una tabla donde colocamos el número a factorizar en la primera fila y vamos dividiendo sucesivamente por primos, registrando cada primo que divide y el cociente resultante hasta llegar a 1.

La Infinitud de los Números Primos

Uno de los teoremas más elegantes de las matemáticas, demostrado por Euclides hace más de 2000 años, **establece que existen infinitos números primos**. La demostración es un ejemplo brillante de razonamiento por contradicción:

Supongamos que **solo existe un número finito de primos**: $p_1, p_2, ..., p_n$. Construimos el número $\mathbf{m = p_1 \times p_2 \times ... \times p_n + 1}$. Este número **m es mayor que 1,** por lo que **debe tener una factorización prima**. **Sin embargo, m no es divisible por ninguno de los primos de nuestra lista finita**, ya que al dividir m por cualquier p_i obtenemos resto 1. **Esto contradice el hecho de que todo número tiene una factorización prima, por lo que nuestra suposición debe ser falsa**.

Aplicaciones Prácticas de los Números Primos

Los números primos tienen aplicaciones fundamentales en criptografía moderna, especialmente en sistemas de clave pública como RSA. La seguridad de estos sistemas se basa en la dificultad computacional de factorizar números grandes que son producto de dos primos grandes. También son esenciales en el diseño de algoritmos de hash, generación de números pseudoaleatorios y en diversas áreas de ciencias de la computación.

Aritmética Modular

Concepto de Congruencia

La aritmética modular introduce una nueva forma de pensar sobre la igualdad entre números. **Decimos que dos números enteros a y b son congruentes**

módulo n (escrito **a ≡ b (mod n)**) **si tienen el mismo resto al dividirlos por n**. Esta idea aparentemente simple crea un sistema aritmético completamente nuevo con propiedades fascinantes.

Por ejemplo, 23 ≡ 58 (mod 7) porque ambos números tienen resto 2 al dividirlos por 7: 23 = 7×3 + 2 y 58 = 7×8 + 21. De manera similar, 15 ≡ 3 (mod 4) porque 15 = 4×3 + 3 y 3 = 4×0 + 3. La congruencia modular nos permite agrupar números en clases de equivalencia basadas en sus restos.

Propiedades Fundamentales de la Congruencia

La relación de congruencia modular satisface tres propiedades fundamentales que la caracterizan como una relación de equivalencia:

Reflexividad: Todo número es congruente consigo mismo módulo cualquier n, es decir, a ≡ a (mod n) para cualquier entero a.

Simetría: Si a ≡ b (mod n), entonces b ≡ a (mod n). Esto significa que la relación de congruencia es bidireccional.

Transitividad: Si a ≡ b (mod n) y b ≡ c (mod n), entonces a ≡ c (mod n). Esta propiedad permite encadenar congruencias.

Estas propiedades garantizan que la congruencia modular divide el conjunto de números enteros en clases de equivalencia disjuntas, donde todos los números en la misma clase tienen el mismo resto al dividirlos por el módulo.

Operaciones en Aritmética Modular

Una de las características más útiles de la aritmética modular es que las operaciones aritméticas básicas se comportan de manera predecible. Si a ≡ b (mod n) y c ≡ d (mod n), entonces:

- **Suma**: a + c ≡ b + d (mod n)
- **Resta**: a - c ≡ b - d (mod n)
- **Multiplicación**: a × c ≡ b × d (mod n)

Estas propiedades permiten realizar cálculos complejos trabajando solo con los restos, lo que simplifica enormemente muchos problemas computacionales.

Las Clases de Equivalencia

Para un módulo n fijo, existen exactamente n clases de equivalencia diferentes, correspondientes a los n posibles restos: 0, 1, 2, ..., n-1. Cada clase contiene todos los números que tienen el mismo resto al dividirlos por n.

Por ejemplo, con módulo 3:

- Clase : {..., -6, -3, 0, 3, 6, 9, ...} (múltiplos de 3)
- Clase : {..., -5, -2, 1, 4, 7, 10, ...} (números de la forma 3k+1)
- Clase : {..., -4, -1, 2, 5, 8, 11, ...} (números de la forma 3k+2)

Aplicaciones Prácticas

La aritmética modular tiene aplicaciones extensas en:

Criptografía: Los algoritmos criptográficos modernos utilizan intensivamente la aritmética modular para crear sistemas seguros de cifrado.

Informática: Los códigos de detección de errores, las funciones hash y los algoritmos de distribución de datos utilizan propiedades de la aritmética modular.

Matemáticas: Facilita el estudio de estructuras algebraicas más complejas como grupos, anillos y campos.

Vida cotidiana: El concepto aparece naturalmente en calendarios (días de la semana, meses), relojes (horas), y sistemas cíclicos en general.

Conclusión

Los tres temas desarrollados en este estudio forman una base sólida para comprender conceptos algebraicos más avanzados. Los sistemas de numeración nos enseñan que la representación de números es flexible y depende del contexto, preparándonos para pensar abstractamente sobre las matemáticas. Los números primos revelan la estructura fundamental subyacente de todos los números naturales, mostrando cómo la complejidad emerge de elementos simples. La aritmética modular introduce una nueva perspectiva sobre las operaciones aritméticas que será esencial para estudios posteriores en álgebra abstracta.

Estos conceptos están profundamente interconectados: la factorización prima utiliza implícitamente diferentes bases para entender la estructura de los números, mientras que la aritmética modular proporciona herramientas para trabajar eficientemente con las propiedades de divisibilidad que definen los

números primos. Dominar estos fundamentos es crucial para avanzar en el estudio del álgebra y sus aplicaciones en ciencias e ingeniería.

El dominio de estos conceptos requiere práctica constante con ejemplos y problemas variados. Se recomienda trabajar extensivamente con conversiones entre bases, factorizaciones primas de números de diferentes tamaños, y operaciones en diferentes módulos para desarrollar intuición matemática sólida. La comprensión profunda de estos temas facilitará enormemente el aprendizaje de estructuras algebraicas más complejas en cursos posteriores.

Apuntes Pedro Lopez Galán