FACULTAD DE INGENIERÍA

Carrera de Ingeniería de Sistemas de Información

Estructuras Discretas II

PRIMER RESUMEN

JOSEPH ALEXANDER CHOMBO MOROCHO

Profesor: Javier Antonio Dioses Zarate

Lima - Perú

2023

Índice general

Semana 1

Fundamentos de la Lógica Proposicional

1.1. Introducción a la Lógica Proposicional

Introducción

Hoy en día la lógica proposicional es un pilar clave en varios campos. En informática, gobierna la programación y la Inteligencia Artificial. En filosofía, construye argumentos sólidos y convincentes. En matemáticas y decisiones, es la base del razonamiento. En resumen, se puede decir que la lógica proposicional es esencial para comprender y resolver problemas en diversos ámbitos.

1.2. ¿Qué es la Lógica Proposicional?

Definición

La lógica proposicional es un sistema que nos permite comprender el universo de las proposiciones y enunciados. A partir de esta base, podemos explorar y clasificar las diversas formas en que el lenguaje se utiliza para comunicar información y expresar ideas. Dos categorías importantes en este contexto son las expresiones enunciativas y las expresiones no enunciativas.

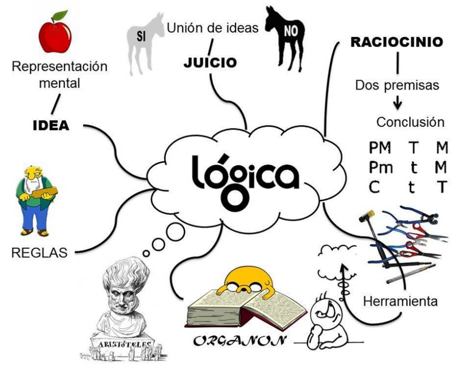


Figura 1.1: *Lógica proposicional*

Ahora, es esencial comprender las dos categorías fundamentales: las expresiones enunciativas y las expresiones no enunciativas.

Expresiones Enunciativas

Una expresión enunciativa es una declaración que tiene un valor de verdad definido, es decir, puede ser evaluada como verdadera o falsa.

Ejemplos:

- Proposición: ***"El cielo es azul."***

  Es una expresión enunciativa que puede ser evaluada como verdadera o falsa en función de si el cielo realmente está azul en un momento dado.

- Proposición: ***"Los pájaros vuelan"***

  Esta proposición es enunciativa ya que puede ser verificada en términos de su verdad en la realidad.

□

Expresiones No Enunciativas

Las expresiones no enunciativas son aquellas que no pueden ser catalogadas como verdaderas o falsas, ya que no están diseñadas para afirmar algo sobre el mundo real, estas pueden ser funciones,deseos, emitir órdenes o formular preguntas.

Ejemplos:

- Expresión: ***"¡Qué bonito!"***

  Esta expresión es emocional y no hace una afirmación concreta. En el contexto de la lógica proposicional no es una proposición con un valor de verdad definido.

- Expresión: ***“¡Salud!”***

  Similar al ejemplo anterior, esta expresión no enunciativa es una interjección que no puede ser tratada directamente como una proposición lógica con un valor de verdad.

□

1.2.1. Conectivas Lógicas

Concepto

Una conectiva lógica es un símbolo que conecta afirmaciones en lógica. Actúa como un puente entre declaraciones, y su función es determinar si la declaración compuesta es verdadera o falsa en función de las declaraciones individuales. Estas conectivas son fundamentales en la construcción de argumentos lógicos y en la programación.

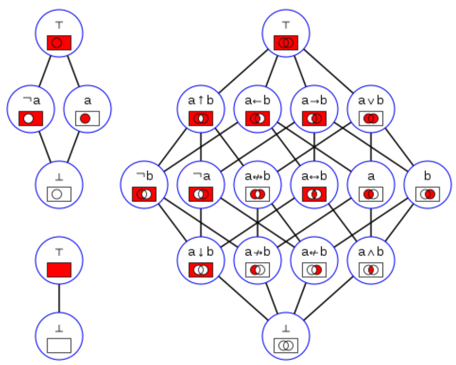


Figura 1.2: *Conectivas lógicas*

A continuación se ve una tabla de las conectivas lógicas mas usadas incluyendo su lenguaje natural y sus respectivos símbolos:

Conectiva	Símbolo	Lenguaje Natural
Negación	$\neg$	no, ni, salvo, excepto
Conjunción	$\wedge$	y, e, o ni, pero, empero, sin embargo, además, aunque, así mismo, sino que, en cambio
Disyunción	$\vee$	o, o bien, o también, o incluso, o hasta
Disyunción Exclusiva	$\oplus$	o. . . o, o uno. . . o el otro, o bien. . . o bien
Condicional	$\rightarrow$	si. . . entonces; por tanto; en consecuencia; siempre que; es suficiente
Bicondicional	$\leftrightarrow$	si y solo si; es necesario y suficiente; ser el mismo que, es necesario que y solo es necesario que

1.2.2. Tablas de Verdad

Una tabla de verdad proporciona una representación visual de cómo las expresiones lógicas se transforman en función de todas las combinaciones posibles de valores de verdad de sus componentes. Cada conectiva lógica tiene su propia tabla, en la cual se muestran los valores de verdad resultantes, generalmente representados por (0) y (1), para cada escenario posible. Estas tablas permiten analizar cómo cada operación lógica impacta en la veracidad global de una expresión, lo que a su vez ayuda a comprender las implicaciones y relaciones lógicas entre proposiciones.

Negación

p	**$\neg$ p**
V	F
F	V

Conjunción

p	**q**	**p $\wedge$ q**
V	V	V
V	F	F
F	V	F
F	F	F

Disyunción

p	**q**	**p $\vee$ q**
V	V	V
V	F	V
F	V	V
F	F	F

Disyunción Exclusiva

p	**q**	**p $\oplus$ q**
V	V	F
V	F	V
F	V	V
F	F	F

Condicional

p	**q**	**p $\rightarrow$ q**
V	V	V
V	F	F
F	V	V
F	F	V

Bicondicional

p	**q**	**p $\leftrightarrow$ q**
V	V	V
V	F	F
F	V	F
F	F	V

1.3. Clasificación

En la lógica tenemos tres estados para evaluar afirmaciones: **tautología, contradicción y contingencia**. Estos estados nos ayudan a entender cómo las proposiciones Sominskii (1959/1961) interactúan con los valores de verdad y cómo algunas afirmaciones son siempre verdaderas, otras siempre falsas, y algunas varian entre ambas posibilidades.

1.3.1. Tautología

Una tautología es una afirmación que, en cualquier circunstancia, resulta verdadera. Es como una **verdad universal** que nunca puede equivocarse, sin importar las condiciones que se apliquen. En esencia, es una afirmación que no puede ser falsa.

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

1.3.2. Contradicción

En el lado opuesto, una contradicción es una afirmación que siempre es falsa. No importa cuánto tratemos de ajustar las condiciones, nunca encontraremos un escenario donde esta afirmación sea verdadera. Es una especie de **falsedad inmutable.**

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

1.3.3. Contigencia

Las afirmaciones de contingencia son un poco más flexibles. Pueden ser verdaderas o falsas dependiendo de cómo se alineen los valores de verdad de sus partes. No son extremadamente rígidas como las tautologías ni extremadamente negativas como las contradicciones. Se ajustan y cambian según las circunstancias.

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

1.4. Corolarios

Corolario 1.4.1 **Leyes de Morgan**

- $(\neg(A \wedge B)) \Leftrightarrow ((\neg A) \vee (\neg B))$
- $(\neg(A \vee B)) \Leftrightarrow ((\neg A) \wedge (\neg B))$

Corolario 1.4.2 **Leyes Asociativas**

- $(A \wedge (B \wedge C)) \Leftrightarrow ((A \wedge B) \wedge C)$
- $(A \vee (B \vee C)) \Leftrightarrow ((A \vee B) \vee C)$

Corolario 1.4.3 **Leyes Conmutativas**

- $(A \wedge B) \Leftrightarrow (B \wedge A)$
- $(A \vee B) \Leftrightarrow (B \vee A)$

Corolario 1.4.4 **Leyes Distributivas**

- $(A \wedge (B \vee C)) \Leftrightarrow ((A \wedge B) \vee (A \wedge C))$
- $(A \vee (B \wedge C)) \Leftrightarrow ((A \vee B) \wedge (A \vee C))$
- $((A \wedge B) \vee C) \Leftrightarrow ((A \vee C) \wedge (B \vee C))$
- $((A \vee B) \wedge C) \Leftrightarrow ((A \wedge C) \vee (B \wedge C))$

Semana 2

Diseño de Lógica Combinacional

2.1. Álgebra Booleana

Definición

Álgebra de Boole, también conocida como álgebra booleana, es una rama del álgebra matemática que se centra en el estudio y manipulación de objetos abstractos llamados "variables booleanas". Estas variables pueden tomar solo dos valores posibles: verdadero (1) o falso (0). Las operaciones fundamentales en el álgebra de Boole son la conjunción (AND), la disyunción (OR) y la negación (NOT)

2.2. Operaciones Básicas

2.2.1. Disyunción (Suma Lógica)

Denominada también operación .º"(OR). Esta operación responde a la siguiente tabla:

a	b	a + b
1	1	1
1	0	1
0	1	1
0	0	0

Figura 2.1: Tabla Disyunción

2.2.2. Conjunción (Producto Lógico)

Denominada también operación "Y"(AND). Esta operación responde a la siguiente tabla:

a	b	a . b
1	1	1
1	0	0
0	1	0
0	0	0

Figura 2.2: Tabla Conjunción

2.2.3. Negación Lógica

Denominada también operación "N"(NOT). Esta operación responde a la siguiente tabla:

a	$\bar{a}$
1	0
0	1

Figura 2.3: Tabla Negación

2.3. Teoremas

Teorema 2.3.1 Regla del cero y la unidad

a) $x + 0 = x$
b) $x + 1 = 1$
c) $x,0 = 0$
d) $x \cdot 1 = x$

Teorema 2.3.2 Idempotencia 0 potencias iguales

a) $x + x = x$
b) $x \cdot x = x$

Teorema 2.3.3 Involución

$\bar{x} = x$

Teorema 2.3.4 Conmutativa

a) Conmutativa de la + : $x + y = y + x$
b) Conmutativa de la : $x \cdot y = y \cdot x$

Teorema 2.3.5 Asociativa

a) Asociativa de la + : $\quad x + (y + z) = (x + y) + z$
b) Asociativa de la . : $\quad x \cdot (y \cdot z) = (x, y) \cdot z$

Teorema 2.3.6 Distributiva

a) Distributiva de la + : $\quad x + (y \cdot z) = (x + y) \cdot (x + z)$
b) Distributiva de la .: $\quad x \cdot (y + z) = (x, y) + (x, z)$

Teorema 2.3.7 Leyes de absorción

a) $x \cdot (x + y) = x$
b) $x + x \cdot y = x$
c) $\boldsymbol{x \cdot (\overline{x} + y) = x \cdot y}$
d) $x + \bar{x} \cdot y = x + y$
e) $\bar{x} \cdot (x + y) = \bar{x} \cdot y$
f) $\bar{x} + x \cdot y = \bar{x} + y$
g) $(x + y) \cdot (x + \bar{y}) = x$
h) $x \cdot y + \bar{x} \cdot y = y$

Teorema 2.3.8 Teoremas de De Morgan

a) $\overline{xy} = \bar{x} + \bar{y}$
b) $\overline{x + y} = \bar{x} \cdot \bar{y}$
c) $x \cdot y = \overline{\bar{x} + \bar{y}}$
d) $\boldsymbol{x + y = \overline{\overline{x}} \cdot \overline{y}}$

2.4. Términos Canónicos

Término en el cual se encuentra uno de cada uno de los literales de la función. Si el término canónico es un producto, se denominará mintermino. Si es una suma se denominará maxtermino.

Términos Canónico MINTERM
$f(x,y,z) = \bar{x}yz$

Términos Canónico MIAXTERM
$f(x,y,z) = x + \bar{y} + \bar{z}$

2.5. Forma Canónica de Funciones Booleanas

Forma Canónica Suma de Productos
Constituida por términos canónicos productos (minterminos) sumados que aparecen una sola vez
Ejemplo:

$$F(X,Y,Z) = \bar{X}\bar{Y}Z + X\bar{Y}\bar{Z} + X\bar{Y}Z + XY\bar{Z} + XYZ$$

Donde a cada variable de cada mintermino se le asocia un valor binario:
A las complementadas: 0
A las no complementadas: 1

Ejemplo:

$$\bar{X}\bar{Y}\mathbf{Z} = 001$$

$$XYZ = 111$$

□

Forma Canónica Producto de Sumas
Constituida por términos canónicos sumas (maxterminos) multiplicados que aparecen una sola vez.
Ejemplo:

$$F(X,Y,Z) = (X+Y+Z)(X+\bar{Y}+Z)(X+\bar{Y}+\bar{Z})$$

Donde a cada variable de cada maxtermino se le asocia un valor binario:
A las complementadas: 1
A las no complementadas: 0

Ejemplo:

$$(X+\bar{Y}+Z) = 0+1+0$$

$$(X+\bar{Y}+\bar{Z}) = 0+1+1$$

□

Matrix Vectorial

arg1

arg2

Teorema 2.5.1 Argumento
argumento 2

Corolario 2.5.1 **afjksfjjfkef**
ejemllosodjf

Argumento
arg2

Ejemplo 2.5.1 (arg1)
arg2

Argumento
logicaarg3

Argumento

Nota:
arg1

arg1
arg2

arg1: arg2 □

Pregunta 1

Calcular la tabla de verdad de: $(p \to q) \vee (p \to \neg q)$

Solución: asdjoqkwddkadokqjwdjijvnkdmckcksdad

Ghidini y Serafini (2014)

□

Semana 3

Algoritmos y Métodos Computacionales

3.1. Introducción a los Algoritmos

Introducción

Los algoritmos son como recetas para la computadora. Son conjuntos de pasos bien definidos que le indican a la máquina cómo resolver un problema o realizar una tarea específica. Los usamos en la vida cotidiana, como seguir una receta para cocinar o seguir un mapa para llegar a un lugar. En el mundo de la computación, los algoritmos son fundamentales para que los programas funcionen de manera eficiente y efectiva.

3.2. Definición de Algoritmos

Definición

Un algoritmo es una secuencia de pasos detallados y precisos que se siguen para llevar a cabo una tarea o resolver un problema. Cada paso debe estar claramente definido y ser comprensible para la máquina que lo ejecuta. Pueden ser simples o complejos, pero en esencia, son la base de todo lo que hacemos con la tecnología.

3.3. Características

Entradas

Los datos o la información que un algoritmo utiliza para llevar a cabo su tarea.

Salidas

Los resultados o respuestas producidas por el algoritmo después de procesar las entradas.

Precisión
La especialidad y claridad de los pasos en un algoritmo, asegurando sea entendible y sin ambigüedad.

Determinismo
La propiedad que asegura que los resultados intermedios de un algoritmo son siempre los mismos para un conjunto de entradas dado, garantizando consistencia en la ejecución.

Carácter finito
La condición que indica que un algoritmo debe terminar en un número finito de pasos, evitando bucles infinitos.

Salida correcta
La capacidad de un algoritmo para producir la respuesta correcta para todas las entradas válidas.

Generalidad
Un algoritmo debe aplicarse a una variedad amplia de situaciones en lugar de estar limitado a un caso específico.

3.4. Análisis y Eficiencia de Algoritmos

Análisis de Algoritmos

El análisis de algoritmos es el proceso de estimar el tiempo y el espacio y cuantos recursos serán requeridos para ejecutar el algoritmo en diferentes situaciones.

Nota:
Para resolver algoritmos se sugiere suponer que se está a la mitad del algoritmo y parte del problema ya se resolvió.

3.5. Complejidad Computacional y Funciones

La notación "O"proporciona una estimación de cómo crece el tiempo de ejecución o la cantidad de recursos de un algoritmo a medida que el tamaño de la entrada se vuelve muy grande. La notación "O" se utiliza para describir el peor escenario posible en términos de tiempo de ejecución de un algoritmo.

3.5.1. Reglas de la Notación O

Regla de la Suma

Si ejecutamos dos fragmentos de código, la complejidad la define el algoritmo más lento.

Si:

$$A = O(N) \text{ y } B = O(N^2)$$

Entonces:

$$O(N + N^2) \rightarrow O(N^2)$$

Regla del Producto por una Constante

Si dos algoritmos tienen sus complejidades $O(N)$ y $O(kN)$ para alguna constante k, entonces la complejidad de ambas es la misma.

$$O(kN) = O(N)$$

Nota:

Entre mas cercano sea el valor de k a N, mas se acercará la complejidad a ser de orden N^2. En este caso se suele representar como $O(MN)$.

Reglas del Producto

Si dos algoritmos (fragmentos) se ejecutan como parte del otro, la complejidad será el resultado de su multiplicación.

Si:

$$A = O(m) \text{ y } B = O(n)$$

Entonces:

$$O(m \times n)$$

3.5.2. Crecimiento de Funciones

A continuación se muestra una gráfica comparativa que ilustra la complejidad y el tiempo de ejecución de diferentes algoritmos:

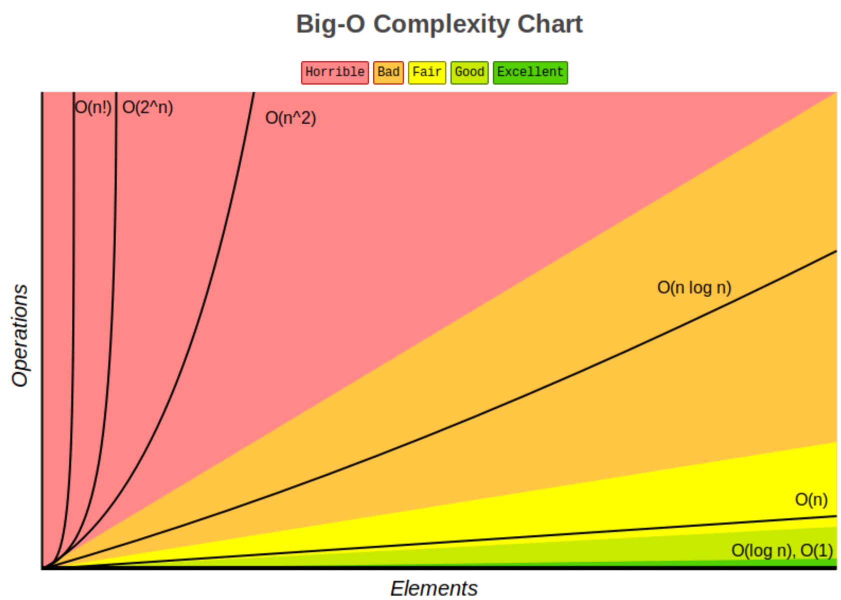


Figura 3.1: Gráfico de complejidad Big O. Fuente: Alarcón (s.f.)

Rendimiento de algunos algoritmos:

Algorithm	Time Complexity			Space Complexity
	Best	Average	Worst	Worst
Quicksort	O(n log(n))	O(n log(n))	O(n^2)	O(log(n))
Mergesort	O(n log(n))	O(n log(n))	O(n log(n))	O(n)
Timsort	O(n)	O(n log(n))	O(n log(n))	O(n)
Heapsort	O(n log(n))	O(n log(n))	O(n log(n))	O(1)
Bubble Sort	O(n)	O(n^2)	O(n^2)	O(1)
Insertion Sort	O(n)	O(n^2)	O(n^2)	O(1)
Selection Sort	O(n^2)	O(n^2)	O(n^2)	O(1)
Shell Sort	O(n)	O((nlog(n))^2)	O((nlog(n))^2)	O(1)

Figura 3.2: Rendimiento de algoritmos y notación Big-O Fuente: Alarcón (s.f.)

3.6. Algoritmos Específicos

3.6.1. QuickSort

QuickSort es un eficiente algoritmo de ordenamiento que sigue el principio de divide y vencerás. Funciona seleccionando un elemento llamado pivote de la lista y particionando los otros elementos en dos sub-arrays según si son menores o mayores que el pivote. Luego, aplica recursivamente el mismo proceso a los sub-arrays. Esto se repite hasta que toda la lista esté ordenada.

Es conocido por su rapidez y eficiencia en la mayoría de los casos, y es ampliamente utilizado en aplicaciones prácticas de ordenamiento de datos.

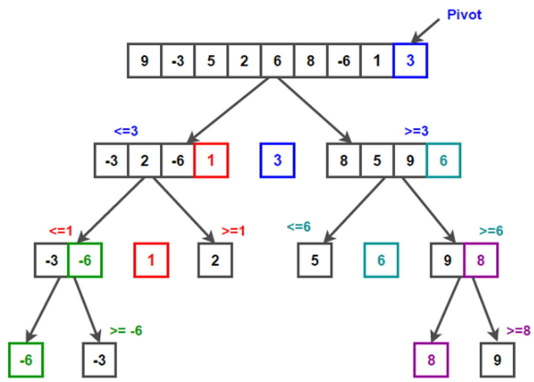


Figura 3.3: Algoritmo Quicksort. Fuente: Didur (2023)

3.6.2. Sucesión de Fibanocci

La sucesión de Fibonacci es una secuencia numérica en la que cada término es la suma de los dos anteriores. Comienza con 0 y 1, y continúa indefinidamente: $0, 1, 1, 2, 3, 5, 8, 13, 21$ y así sucesivamente. Esta secuencia tiene aplicaciones en una amplia variedad de campos, desde matemáticas puras hasta ciencias naturales y computación, y se encuentra en fenómenos naturales y estructuras biológicas. Su patrón recursivo es central en muchos algoritmos y problemas de programación.

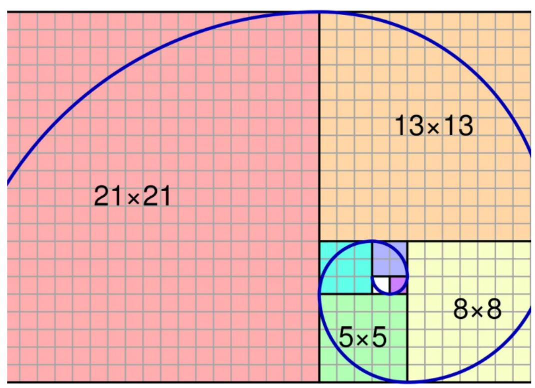


Figura 3.4: Sucesión de Fibonacci. Fuente: Landau (2020)

3.6.3. Algoritmos Recursivos

Los algoritmos recursivos son como un conjunto de instrucciones que, en su ejecución, invocan repetidamente una función que se refiere a sí misma. Esta técnica ofrece una forma poderosa y natural de abordar problemas complejos.

Se basa en el principio de "divide y vencerás", descomponiendo el problema original en subproblemas similares. Cada uno de estos subproblemas se fragmenta aún más hasta que se alcanzan casos base que pueden resolverse directamente. Luego, las soluciones de los subproblemas se combinan de manera estructurada para producir la solución del problema original.

Ejemplo: Este es un algoritmo recursivo para calcular el factorial de un número. □

```
int factorial(n) {
    if (n == 0)
        return 1
    return n * factorial(n - 1)
}
```

3.7. ¿Qué es el Método Montecarlo?

Método Montecarlo

El Método Montecarlo es una técnica probabilística que se basa en la generación de números aleatorios para resolver problemas que pueden no tener una solución determinista o para los cuales es impráctico calcular exhaustivamente todas las posibilidades. Se utiliza ampliamente en áreas como la física, la estadística, la computación y la ingeniería para obtener aproximaciones de soluciones precisas a través de la repetición de experimentos aleatorios. Su nombre proviene del famoso casino de Montecarlo, donde los juegos de azar están gobernados por la probabilidad y el azar.

3.7.1. Aplicaciones del Método Montecarlo

Simulaciones Físicas

En física, se utiliza para simular sistemas complejos y prever su comportamiento en condiciones variadas.

Finanzas

En el ámbito financiero, el Método Montecarlo es esencial para evaluar riesgos y tomar decisiones de inversión.

Estadísticas y Probabilidades

Permite estimar resultados en situaciones donde los cálculos exactos pueden ser impracticables.

Semana 4

Sucesiones e Inducción Matemática

Introducción

En el ámbito de las matemáticas, las sucesiones y la inducción matemática son herramientas fundamentales que nos permiten explorar patrones, demostrar propiedades y resolver problemas complejos. Al combinar estas dos herramientas, somos capaces de abordar una amplia gama de problemas matemáticos con rigor y precisión.

4.1. Sucesiones

Definición

Una sucesión es una secuencia ordenada de elementos que siguen un patrón específico. Las sucesiones pueden ser finitas o infinitas y son fundamentales para estudiar comportamientos y patrones en diversos contextos matemáticos.

4.1.1. Tipo de Sucesiones

Sucesión Alternante

Una sucesión alternante es una secuencia de elementos en la que cada término alterna entre ser positivo y negativo. Esto significa que los elementos de la sucesión van cambiando de signo en un patrón regular.

$$
\begin{aligned}
a_0 &= (-1)^0 = 1, \\
a_1 &= (-1)^1 = -1, \\
a_2 &= (-1)^2 = 1, \\
a_3 &= (-1)^3 = -1, \\
&\vdots
\end{aligned}
$$

Sucesión con Fórmula Explícita

Es una secuencia de números en la que cada término está definido directamente en función de su posición en la secuencia.Proporciona una regla matemática clara que nos permite calcular cualquier término de la sucesión sin necesidad de conocer los términos anteriores.

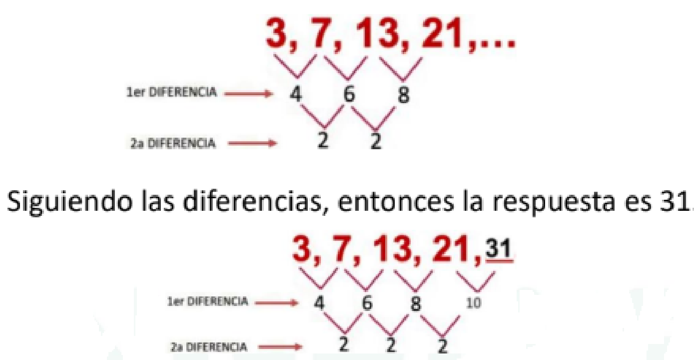
Por ejemplo: La sucesión de números pares

$$2, 4, 6, 8, 10, 12, ...$$

Esta sucesión puede ser expresada de manera explícita como $a_n = 2n$, donde n representa la posición del término en la secuencia □

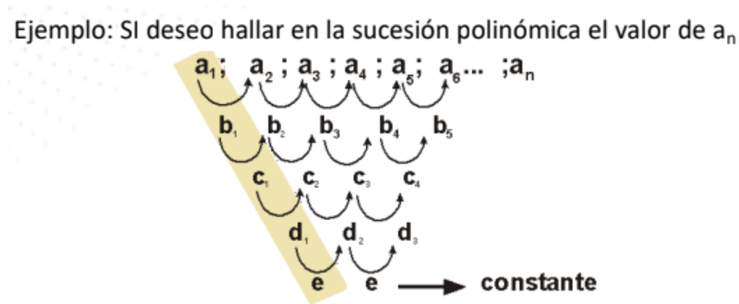
Sucesión Cuadrática

Una sucesión cuadrática es una secuencia de números en la que cada término es una función cuadrática de su posición en la secuencia. Esto significa que la fórmula para calcular el términoa_n de la secuencia está basada en una expresión de la forma $an^2 + bn + c$, donde n representa la posición del término en la secuencia y a, b y c son constantes.



Sucesión Polinomial

Una sucesión polinomial es una secuencia de números en la que cada término se obtiene a partir de una función polinómica de su posición en la secuencia. Esto implica que la fórmula para calcular el término a_n depende de una expresión polinómica en n, donde n representa la posición del término.



Hay que hacer un análisis de la sucesión

$$a_n = a_1 + b_1 c_1^{n-1} + c_1 c_2^{n-1} + d_1 c_3^{n-1} + e_1 c_4^{n-1}$$

El valor de la combinación n sobre k es:

$$C_k^n = \frac{\overbrace{n(n-n)(n-2)\ldots}^{\text{"k"factores}}}{\underbrace{1\times2\times3\times4\times\ldots}_{\text{"k"factores}}} = \frac{n!}{(n-k)!k!}$$

4.2. Inducción Matemática

Definición

La inducción matemática es un método de prueba utilizado para demostrar que una afirmación es cierta para todos los números naturales. Se basa en dos pasos clave: primero, se verifica que la afirmación es cierta para un número inicial (caso base). Luego, se demuestra que si es válida para un número k arbitrario, también lo es para k+1 (paso inductivo). Este método es esencial para establecer propiedades y teoremas que abarcan una infinidad de números enteros.

4.2.1. Método de demostración por inducción matemática

Considere un enunciado de la forma, "Para todo entero $n \geqslant a$, una propiedad $P(n)$ es verdadera". Para demostrar este enunciado, realice los siguientes pasos:
Paso 1 (paso básico): Demuestre que $P(a)$ es verdadera.
Paso 2 (paso inductivo): Demuestre que para todo entero $k \geqslant a$, si $P(k)$ es verdadera entonces $P(k+1)$ es verdadera. Para realizar este paso,
suponga que $P(k)$ es verdadera, donde k es cualquier entero dado pero elegido arbitrariamente con $k \geqslant a$. [Esta suposición se llama hipótesis inductiva.] Después, demuestre que $P(k+1)$ es verdadera.

Por ejemplo: Utilice inducción matemática para demostrar que:

$$1 + 2 + \cdots + n = \frac{n(n+1)}{2}$$

Para construir una demostración por inducción, primero debe identificar la propiedad $P(n)$. En este caso, $P(n)$ es la ecuación:

$$1 + 2 + \cdots + n = \frac{n(n+1)}{2}. \leftarrow \text{ la propiedad } (P(n))$$

Por tanto vamos a verificar si cada uno de estos casos es verdadero:

$$1 = \frac{1(1+1)}{2}. \leftarrow \text{ básico } P(1) \quad \frac{2(2+1)}{2} = \frac{2 \cdot 3}{2} = 3$$

□

4.3. Recurrencia

Concepto

En matemáticas y ciencias de la computación, una recurrencia es una relación o fórmula que define los términos sucesivos de una secuencia utilizando uno o más términos anteriores. Es decir, cada término de la secuencia se calcula en función de los términos previos de la misma secuencia. Las recurrencias son fundamentales en la teoría de sucesiones y desempeñan un papel crucial en áreas como la combinatoria, el álgebra y el análisis de algoritmos. Se utilizan para modelar una amplia variedad de fenómenos y situaciones que implican dependencias temporales o condicionales entre los elementos de una secuencia.

Semana 5

Probabilidades

5.1. ¿Qué son las Probabilidades?

Probabilidades

Las probabilidades son una rama de las matemáticas que se centra en cuantificar y estudiar la incertidumbre asociada a eventos o experimentos. Esta teoría nos permite analizar y predecir la posibilidad de que ciertos resultados ocurran en situaciones donde el resultado no es completamente predecible. Desde los juegos de azar hasta la toma de decisiones en condiciones de incertidumbre, las probabilidades son fundamentales en una amplia variedad de contextos.

5.2. Probabilidad Condicional

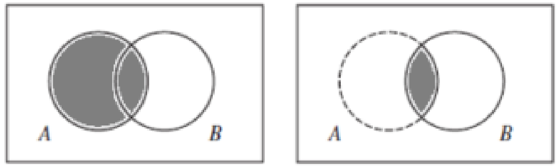
Definición

La probabilidad condicional es una medida de la probabilidad de que ocurra un evento, dado que otro evento ya ha ocurrido. Se denota como $P(A|B)$, que representa la probabilidad de que el evento A ocurra, dado que el evento B ya ha ocurrido.

Teorema 5.2.1 Probabilidad Condicional

Sean A y B eventos con $\boldsymbol{P(B) \neq 0}$. La probabilidad condicional de A dado B es:

$$P(A \mid B) = \frac{P(A \cap B)}{P(B)}$$



P(A) – Probabilidad incondicional   P(A|B) – Probabilidad condicional

5.3. Eventos Independientes

Definición

Dos eventos son considerados independientes si la ocurrencia o no ocurrencia de uno de los eventos no tiene influencia sobre la probabilidad de que ocurra el otro. En otras palabras, la probabilidad de que ambos eventos sucedan es igual al producto de sus probabilidades individuales.

Teorema 5.3.1 Eventos Independientes

Si A y B son eventos independientes, entonces:
1. A y $\bar{B}$ son eventos independientes. 2. $\bar{A}$ y $\bar{B}$ son eventos independientes. 3. $\bar{A} y B$ son eventos independientes.

$$P(\bar{B} \mid A) = \frac{P(A \cap \bar{B})}{P(A)}$$
$$P(A \cap \bar{B}) = P(A) \cdot P(\bar{B} \mid A)$$
$$P(A \cap \bar{B}) = P(A) \cdot (1 - P(B \mid A))$$
$$P(A \cap \bar{B}) = P(A) \cdot (1 - P(B))$$
$$P(A \cap \bar{B}) = P(A) \cdot P(\bar{B})$$

5.3.1. Reglas de Multiplicación

Cuando dos eventos son independientes se cumple:

$$\mathrm{P}(A \mid B) = P(A)$$
$$\mathrm{P}(B \mid A) = P(B)$$

Reemplazando en la regla de multiplicación:

$$P(A \cap B) = P(A)P(B \mid A) \text{ ó } P(A \cap B) = P(B)P(A \mid B)$$

Se simplifica en:

$$P(A \cap B) = P(A)P(B)$$

El resultado se puede ampliar para cualquier número de eventos

Si $A_1, A_2, \ldots\ldots\ldots A_n$ siendo eventos independientes, entonces se tiene que para cada colección $A_{j1}, A_{j2}, \ldots\ldots A_{jm}$ de eventos.

$$P\left(A_1 \cap A_2 \cap \cdots \cap A_n\right) = P\left(A_1\right) P\left(A_2\right) \cdots P\left(A_n\right)$$

5.4. Probabilidad Total

Definición

La probabilidad total es un concepto fundamental en teoría de probabilidad que se refiere a la probabilidad de un evento A que puede ocurrir a través de diferentes escenarios o çaminos"mutuamente excluyentes. Se calcula como la suma de las probabilidades condicionales de que ocurra el evento A dadas las diferentes situaciones posibles, multiplicadas por la probabilidad de que ocurra cada una de esas situaciones.

Teorema 5.4.1 Probabilidad Total

Como $B = (A_1 \cap B) \cup (A_2 \cap B) \cup (A_3 \cap B) \cup (A_4 \cap B)$, es la unión de eventos mutuamente excluyentes, se obtiene que:

$$P(B) = P\left(A_1 \cap B\right) + P\left(A_2 \cap B\right) + P\left(A_3 \cap B\right) + P\left(A_4 \cap B\right)$$

Y tomando en cuenta la regla de probabilidad condicional:

$$P\left(A_i \cap B\right) = P\left(B \mid A_i\right) P\left(A_i\right)$$

Se tiene:

$$P(B) = P\left(B \mid A_1\right) P\left(A_1\right) + P\left(B \mid A_2\right) P\left(A_2\right) + P\left(B \mid A_3\right) P\left(A_3\right) + P\left(B \mid A_4\right) P\left(A_4\right)$$

5.5. Regla de Bayes

Definición

Es un principio fundamental en teoría de probabilidad y estadística. Se utiliza para actualizar creencias o probabilidades sobre un evento, dado cierta evidencia o información adicional. La regla de Bayes establece cómo combinar la probabilidad de que ocurra un evento con la probabilidad de que la evidencia observada sea consistente con ese evento. Es especialmente valiosa en situaciones donde la información disponible es incompleta o incierta.

Teorema 5.5.1 Teorema de Bayes

Se tiene $\mathrm{P}(\mathrm{A} \mid \mathrm{B}) \neq \mathrm{P}(\mathrm{B} \mid \mathrm{A})$ y se desea calcular uno de las probabilidades condicionales, la regla de Bayes proporciona una fórmula que permite calcular una de las probabilidades condicionales si se conoce la otra. Se tiene:

$$\mathrm{P}(\mathrm{A} \mid \mathrm{B}) = \frac{P(A \cap B)}{P(B)}$$

$$\mathrm{P}(\mathrm{A} \mid \mathrm{B}) = \frac{P(B|A)P(A)}{P(B)}$$

Semana 6

Grafos

6.1. ¿Qué son los Grafos?

Introducción

En el ámbito de las matemáticas y la teoría de redes, los grafos son una abstracción poderosa utilizada para representar y analizar relaciones entre distintos elementos. Esta estructura se compone de nodos, que representan entidades individuales, y aristas que denotan las conexiones o interacciones entre estos nodos. Los grafos proporcionan un marco conceptual esencial para comprender una amplia gama de fenómenos, desde redes sociales hasta sistemas de transporte y distribución. Su versatilidad y aplicaciones en áreas como la informática, la logística y la biología hacen de los grafos un concepto fundamental en diversas disciplinas.

6.2. Grafos

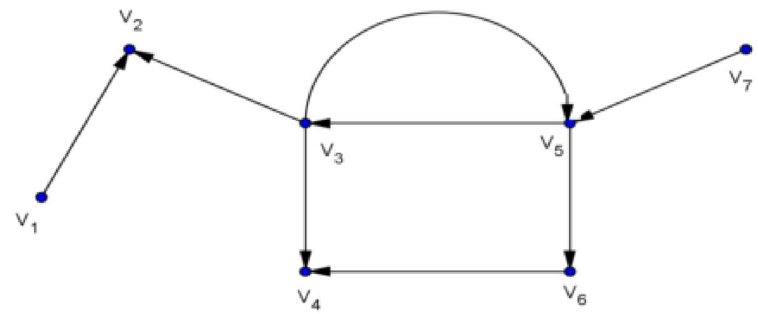
Definición

Formalmente, un grafo G se define como un par ordenado$G = (V, E)$, donde V es un conjunto de nodos o vértices, y E es un conjunto de aristas o arcos. Cada arista conecta un par de nodos y puede ser dirigida (con orientación) o no dirigida (sin orientación). Los grafos pueden ser simples, donde no hay aristas múltiples que conecten el mismo par de nodos, donde esto es permitido.

6.2.1. Tipos de Grafos

Camino Elemental

Es el camino que nunca pasa más de una vez por un mismo punto.

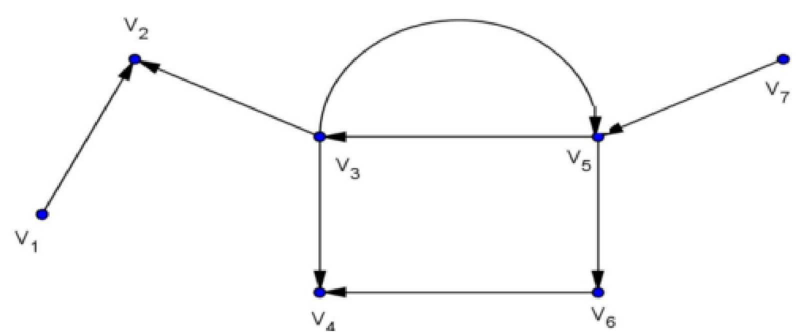


Estos son los caminos elementales:

$$\gamma(v_1, v_4) = \gamma(v_1, v_2, v_3, v_4) \quad \text{ya que} \quad \gamma(v_1, v_4) = \gamma(v_1, v_2, v_3, v_5, v_6, v_4)$$

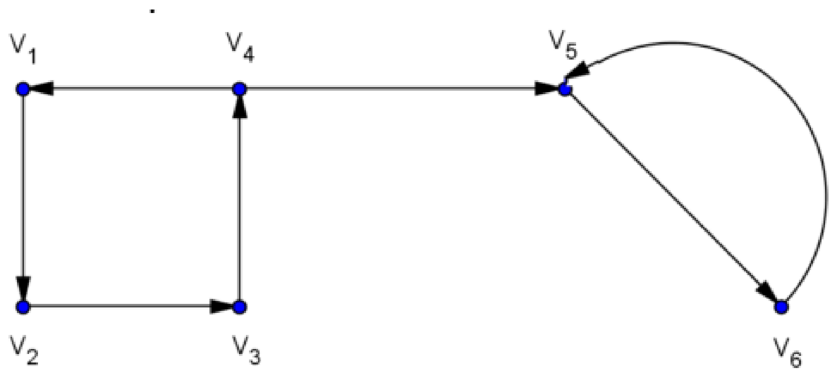
Pero no es un camino elemental, porque pasa dos veces por el v_3: $\gamma(v_1, v_2, v_3, v_5, v_3, v_4)$

Camino Compuesto
Es aquel camino que utiliza más de una vez un mismo punto.



Es un camino compuesto: $\gamma\,(v_1, v_2, v_3, v_5, v_3)$

Camino Circuito
Es aquel camino que vuelve a su punto de origen. Todo punto de partida es a la vez punto de llegada. Ejemplo



6.3. Tipos de Búsquedas

6.3.1. Búsqueda Voraz

Expande el nodo más cercano al objetivo.
Evalúa los nodos utilizando solamente la función heurística, que, en general, se minimiza, porque se refiere a un coste:
• f(n) = h(n)
• La minimización de h(n) es susceptible de ventajas falsas.

6.3.2. Búsqueda A*

La búsqueda A* es un algoritmo de búsqueda informada que se utiliza para encontrar el camino más corto en un grafo ponderado dirigido o no dirigido. Combina la eficiencia de la búsqueda voraz con la capacidad de encontrar la solución óptima. El algoritmo evalúa los nodos en función de dos valores: el costo acumulado desde el nodo de inicio hasta el nodo actual (g(n)), y una estimación del costo desde el nodo actual hasta el nodo objetivo ($h(n)$). La función de evaluación $f(n) = g(n) + h(n)$ guía la elección del próximo nodo a expandir, eligiendo el nodo con el menor valor de $f(n)$.

Semana 7

Recursos Complementarios

7.1. Semana 1

Pregunta 2

Calcular la tabla de verdad de: $(p \longrightarrow q) \vee (p \longrightarrow \neg q)$

Solución:

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

Ghidini y Serafini (2014)

□

Pregunta 3

Calcular la tabla de verdad de: $(p \longrightarrow q) \vee (p \longrightarrow \neg q)$

Solución:

p	**q**	**r**	**(¬ p**	∨	**q)**	∧	**(q**	→	¬ **r**	∧	¬ **p)**	∧	**(p**	∨	**r)**
V	V	V	F	**V**	V	F	V	F	F	**F**	F	F	V	V	V
V	V	F	F	**V**	V	F	V	F	V	**F**	F	F	V	V	F
V	F	V	F	**F**	F	F	F	V	F	**F**	F	F	V	V	V
V	F	F	F	**F**	F	F	F	V	V	**F**	F	F	V	V	F
F	V	V	V	**V**	V	F	V	F	F	**F**	V	F	F	V	V
F	V	F	V	**V**	V	V	V	V	V	**V**	V	F	F	F	F
F	F	V	V	**V**	F	V	F	V	F	**F**	V	V	F	V	V
F	F	F	V	**V**	F	V	F	V	V	**V**	V	F	F	F	F

Ghidini y Serafini (2014)

□

7.2. Semana 2

7.3. Semana 3

Pregunta 4

Crea un método que calcule el número de fibonacci a partir de un número pasado como parámetro.

Solución:

Aulanuvbe (2022)

□

```
public static void main(String[] args)
{
    //numero de fibonacci
    int num = 5;

    System.out.println("Fibonacci de "+num+" es "+fibonacci1(num));
    System.out.println("Fibonacci de "+num+" es "+fibonacci2(num));
    System.out.println("Fibonacci de "+num+" es "+fibonacci3(num));
}
static int fibonacci1(int n)
{
    if (n>1)  return fibonacci1(n-1) + fibonacci1(n-2);
```

```
    else if (n==1)
        return 1;
    else
        return 0;
}
```

7.4. Semana 4

Pregunta 5

Probar que: $1 + x + x^2 + \vdash \ldots + x^n = \frac{x^{n+1}-1}{x-1} \quad (x \neq 1)$.

Solución:

1) Para $n = 1$ la proposición es verdadera.

2) Se tiene:

$$1 + x + x^2 + \ldots + x^k = \frac{x^{k+1}-1}{x-1}$$

Entonces:

$$1 + x + x^2 + \ldots + x^k + x^{k+1} = \frac{x^{k+1}-1}{x-1} + x^{k+1} = \frac{x^{k+2}-1}{x-1}$$

Sominskii (1959/1961)

□

7.5. Semana 5

Pregunta 6

El 76 % de los estudiantes de Ingeniería Civil han aprobado resistencia de materiales y el 45 % aprobaron estática. Además, el 30 % aprobaron resistencia de materiales y estática. Si Camilo aprobó resistencia de materiales, ¿qué probabilidad tiene de haber aprobado también estática?

Solución: Vamos a trabajar con 2 eventos: aprobar resistencia de materiales, y aprobar estática.- Evento A : aprobar resistencia de materiales. $\mathbf{P(A) = 76}$ %.- Evento B: aprobar estática. $\mathbf{P(B) = 45}$ %.- Evento A y B: aprobar resistencia de materiales y estática. $\mathbf{P(A \cap B) = 30}$ %, y es lo mismo que: $\mathbf{P(B \cap A) = 30}$ %Ahora calculamos la probabilidad de aprobar estática, dado que se aprobó resistencia de materiales.

$$P(B \mid A) = \tfrac{P(B \cap A)}{P(A)}$$
$$P(B \mid A) = \tfrac{30\,\%}{76\,\%} = \tfrac{30}{76} = \tfrac{15}{38} = 0,3947 = 39,47\,\%$$

Para Camilo, la probabilidad de aprobar estática, dado que aprobó resistencia de materiales es de 39,47 %.

J. y (2021)

□

7.6. Semana 6

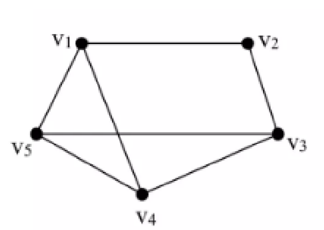
Pregunta 7

Dado un grafo con matriz de adyacencia

$$\begin{pmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \end{pmatrix}$$

A) El grafo es euleriano
B) El grafo es conexo
C) Es un multigrafo

Solución: Si graficamos tenemos:



Es conexo, dado que no hay ningún vértice aislado de los demás.No es euleriano porque los grados de los vértices son: 3, 2, 3, 3, 3. (Para que sea grafo euleriano tienen que ser todos los grados pares).No es multigrafo porque no hay todos los números que aparecen son 0 ó 1 . Para que sea multigrafo propiamente deben existir más un número superior a 1 , indicando con ellos que dos vértices se conectan con más de una arista.

Fernandez (s.f.)

□

Bibliografía

Alarcón, J. M. (s.f.). Rendimiento de algoritmos y notación Big-O - campusMVP.es. https://www.campusmvp.es/recursos/post/Rendimiento-de-algoritmos-y-notacion-Big-O.aspx

Aulanuvbe. (2022, 19 de noviembre). *Ejercicios resueltos de recursividad en JAVA*. https://aulaenlanube.com/zona-programacion/java/ejercicios-recursividad-java/

Didur, M. (2023). Parallel Quick Sort on C++ - Mykhailo Didur - Medium. https://medium.com/@lucky_rydar/parallel-quick-sort-on-c-a3526ba8b532

Fernandez, T. (s.f.). *Ejercicios resueltos grafos*. https://www.slideshare.net/terfer/ejercicios-resueltos-grafos-149128459

Ghidini, C., & Serafini, L. (2014). *Mathematical Logic Exercises* [Anno Accademico 2013-2014. We thank Annapaola Marconi for her work in previous editions of this booklet.].

, J., &, J. (2021). Probabilidad condicional, ejercicios resueltos | Matemóvil. *MateMovil*. https://matemovil.com/probabilidad-condicional-ejercicios-resueltos/

Landau, E. (2020). The Fibonacci sequence is Everywhere—Even the troubled stock market. https://www.smithsonianmag.com/science-nature/fibonacci-sequence-stock-market-180974487/

Sominskii, I. S. (1961). *Metod matematicheskoi induktsii* [The Method of Mathematical Induction] (H. Moss & I. N. Sneddon, Trad.) [Library of Congress Card Number: 61-11532]. Blaisdell Publishing Company. (Fecha inicial de publicación 1959)