MATEMÁTICA DISCRETA

Índice

[Unidad 1: Lógica y teoría de conjuntos 2](#_Toc279796379)

[*1.* *Definiciones* 2](#_Toc279796380)

[*2.* *Leyes de la lógica* 2](#_Toc279796381)

[*3.* *Reglas de inferencia* 3](#_Toc279796382)

[*4.* *Lógica de predicados* 3](#_Toc279796383)

[*5.* *Teoría de conjuntos* 3](#_Toc279796384)

[Unidad 2: Inducción matemática 4](#_Toc279796385)

[*1.* *Métodos para demostrar la verdad de una implicación* 4](#_Toc279796386)

[*2.* *Inducción matemática* 4](#_Toc279796387)

[Unidad 3: Relaciones de recurrencia 4](#_Toc279796388)

[*1.* *Ecuaciones de recurrencia homogéneas* 5](#_Toc279796389)

[*2.* *Ecuaciones de recurrencia no homogéneas* 5](#_Toc279796390)

[*3.* *Sucesiones importantes* 5](#_Toc279796391)

[Unidad 4: Relaciones 6](#_Toc279796392)

[*1.* *Definiciones* 6](#_Toc279796393)

[*2.* *Propiedades de las relaciones* 6](#_Toc279796394)

[*3.* *Matriz de una relación* 6](#_Toc279796395)

[*4.* *Relaciones de equivalencia y de orden* 6](#_Toc279796396)

[*5.* *Elementos particulares* 7](#_Toc279796397)

[Unidad 5: Álgebras de Boole 7](#_Toc279796398)

[*1.* *Definiciones y axiomas* 7](#_Toc279796399)

[*2.* *Funciones booleanas* 8](#_Toc279796400)

[*3.* *Propiedades de los átomos* 9](#_Toc279796401)

[*4.* *Mapa de Karnaugh* 9](#_Toc279796402)

[*5.* *Isomorfismos entre álgebras de Boole* 10](#_Toc279796403)

[Unidad 6: Teoría de grafos 10](#_Toc279796404)

[*1.* *Definiciones de grafos y digrafos* 10](#_Toc279796405)

[*2.* *Aristas, vértices, caminos y grafos* 10](#_Toc279796406)

[*3.* *Grafos de Euler* 12](#_Toc279796407)

[*5.* *Representación de grafos por matrices* 13](#_Toc279796408)

[*6.* *Niveles* 14](#_Toc279796409)

[*7.* *Algoritmos de camino mínimo* 14](#_Toc279796410)

[Unidad 7: Árboles 15](#_Toc279796411)

[*1.* *Definiciones* 15](#_Toc279796412)

[*2.* *Árboles generadores* 16](#_Toc279796413)

[*3.* *Algoritmos para hallar un árbol generador mínimo* 16](#_Toc279796414)

[Unidad 8: Redes de transporte 16](#_Toc279796415)

[*1.* *Definiciones* 16](#_Toc279796416)

[*2.* *Algoritmo de Ford-Foulkerson* 17](#_Toc279796417)

# **Unidad 1: Lógica y teoría de conjuntos**

## Definiciones

Lógica: estudio de las formas correctas de pensar o razonar.

Proposición: afirmación que es verdadera o falsa, pero no ambas.

Proposición primitiva: proposición que no se puede descomponer en otras dos o más proposiciones. Siempre son afirmativas.

Proposición compuesta**:** proposición formada por dos o más proposiciones relacionadas mediante conectivas lógicas.

Tablas de verdad:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **p** | **q** | ¬**p (NOT)** | **p q (AND)** | **p q (OR)** | **p q (XOR)** | p → q (IF) | **p ↔ q (IIF)** | **p ↓ q (NOR)** | **p | q (NAND)** |
| V | V | F | V | V | F | V | V | F | F |
| V | F | F | F | V | V | F | F | F | V |
| F | V | V | F | V | V | V | F | F | V |
| F | F | V | F | F | F | V | V | V | V |

*Nota*: proposiciones → líneas de tabla.

Negación: no, nunca, jamás, no es cierto que.

Conjunción: y, e, pero, como, aunque, sin embargo, mientras.

Disyunción: o, a menos que.

Disyunción excluyente: o bien.

Implicación: cuando, siempre que.

Doble implicación: si y sólo si (sii), cuando y solo cuando.

{|} y {↓} son los únicos conjuntos adecuados de un solo conectivo diádico.

|  |  |
| --- | --- |
| **“p q”** | **“p q”** |
| * Si p, entonces q. * p implica q. * p solo si q. * p es el antecedente, q es el consecuente. * q es necesario para p. * p es suficiente para q. | * p es necesario y suficiente para q. * p si y solo si q. |

Tautología**:** proposición que es verdadera siempre.

Contradicción**:** proposición que es falsa siempre.

Contingencia**:** proposición que puede ser verdadera o falsa, dependiendo de los valores de las proposiciones que la componen.

* p **→** q ≡ ¬p ∨ q
* p ↔ q ≡ (p **→** q) ∧ (q **→** p)
* (p ∨ q) ≡ (p ∨ q) ∧ (¬p ∨ ¬q)
* a **→** (b ∧ c) ≡ (a **→** b) ∧ (a **→** c)
* (p ∨ q) **→** t ≡ (p **→** t) ∨ (q **→** t)

## Leyes de la lógica

|  |  |
| --- | --- |
| 1. Ley de la doble negación | ¬¬p ≡ p |
| 1. Ley de conmutatividad | 1. p ∨ q ≡ q ∨ p 2. p ∧ q ≡ q ∧ p |
| 1. Ley de asociatividad | 1. p ∨ (q ∨ r) ≡ (p ∨ q) ∨ r 2. p ∧ (q ∧ r) ≡ (p ∧ q) ∧ r |
| 1. Ley de distributividad | 1. p ∧ (q ∨ r) ≡ (p ∧ q) ∨ (p ∧ r) 2. p ∨ (q ∧ r) ≡ (p ∨ q) ∧ (p ∨ r) |
| 1. Ley de idempotencia | 1. p ∨ p ≡ p 2. p ∧ p ≡ p |
| 1. Ley del elemento neutro | 1. p ∨ F0 ≡ p 2. p ∧ T0 ≡ p |
| 1. Leyes de De Morgan | 1. ¬(p ∨ q) ≡ ¬p ∧ ¬q 2. ¬(p ∧ q) ≡ ¬p ∨ ¬q |
| 1. Ley del inverso | 1. p ∨ ¬p ≡ T0 2. p ∧ ¬p ≡ F0 |
| 1. Ley de dominancia | 1. p ∨ T0 ≡ T0 2. p ∧ F0 ≡ F0 |
| 1. Ley de absorción | 1. p ∨ (p ∧ q) ≡ p 2. p ∧ (p ∨ q) ≡ p |

Dual de S: Sea S una proposición. Si S *no contiene* conectivas lógicas distintas de ∧ y ∨ entonces el dual de S (Sd), se obtiene de reemplazar en S todos los ∧ (∨) por ∨ (∧) y todas las T0 (F0) por F0 (T0).

Sean s y t dos proposiciones tales que s ≡ t, entonces sd ≡ td.

* *Recíproca:*(q **→** p) es la recíproca de (p **→** q)
* *Contra-recíproca:* (¬q **→** ¬p) es la contra-recíproca de (p **→** q)
* *Inversa:*(¬p **→** ¬q) es la inversa de (p **→** q)

## Reglas de inferencia

|  |  |
| --- | --- |
| **Modus ponens o Modus ponendo ponens** | p **→** q  p  ∴ q |
| **Modus tollens o Modus tollendo tollens** | p **→** q  ¬q  ∴ ¬p |

## Lógica de predicados

Función proposicional: expresión que contiene una o más variables que al ser sustituidas por elementos del universo dan origen a una proposición.

Universo: Son las ciertas opciones “permisibles” que podré reemplazar por la variable.

Cuantificador universal: proposición que es verdadera para todos los valores de en el universo.

Cuantificador existencial: proposición en que existe un elemento del universo tal que la función proposicional es verdadera.

**Negación de proposiciones cuantificadas:**

* ¬[∀x p(x)] ≡ ∃x ¬p(x)
* ¬[∃x p(x)] ≡ ∀x ¬p(x)
* ∃x [p(x) ∧ q(x)] ⇒ ∃x p(x) ∧ ∃x q(x)
* ∃x [p(x) ∨ q(x)] ⬄ ∃x p(x) ∨ ∃x q(x)
* ∀x [p(x) ∧ q(x)] ⬄ ∀x p(x) ∧ ∀x q(x)
* ∀x p(x) ∨ ∀x q(x) ⇒ ∀x [p(x) ∨ q(x)]
* ∃x [p(x) ∧ q(x)] ≠ ∃x p(x) ∧ q(x)

## Teoría de conjuntos

Conjunto de partes: dado un conjunto A, p(A) es el conjunto formado por todos los subconjuntos de A, incluídos A y . Si A tiene elementos, p(A) tendrá elementos. *Ejemplo*:

Pertenencia: un elemento “pertenece” a un conjunto.

Inclusión: un conjunto está “incluido” en un conjunto.

**Operaciones entre conjuntos:**

Unión:

Intersección:

Diferencia:

Diferencia simétrica:

Complemento:

##### Leyes del álgebra de conjuntos: Para cualquier A, B ⊆ U:

|  |  |
| --- | --- |
| Leyes conmutativas |  |
| Leyes asociativas |  |
| Leyes distributivas |  |
| Leyes de idempotencia |  |
| Leyes de identidad |  |
| Complementación doble |  |
| Leyes del complemento |  |
| Leyes de De Morgan |  |

# **Unidad 2: Inducción matemática**

## Métodos para demostrar la verdad de una implicación

1. Método directo:V **→** V
2. Método indirecto:
3. Por el contrarrecíproco: F ← F
4. Por el absurdo: supongo el antecedente verdadero y el consecuente falso y busco llegar a una contradicción de proposiciones.

## Inducción matemática

# **Unidad 3: Relaciones de recurrencia**

Orden de una relación: mayor subíndice – menor subíndice.

## Ecuaciones de recurrencia homogéneas

Sea la ecuación (\*). Resolverla significa:

1. Hallar las raíces de la ecuación característica de (\*):
2. Utilizar los teoremas siguientes para hallar la solución.

**Teorema 1**: si y son soluciones de la ecuación (\*), entonces también es solución de (\*) .

**Teorema 2**: si es raíz de la ecuación característica, entonces es solución de (\*).

**Teorema 3**: si y () son soluciones de la ecuación característica, entonces es solución de (\*)y

**Teorema 4**: si es raíz doble de la ecuación característica, entonces es solución de (\*).

**Teorema 5**: si es raíz doble de la ecuación característica, entonces es solución de (\*) y

## Ecuaciones de recurrencia no homogéneas

Sea la ecuación (\*), con . Resolverla significa:

1. Resolver la ecuación homogénea asociada y obtener .
2. Hallar *una* solución particular de la ecuación (\*), .
3. La solución general será:

*Nota*: en la solución particular propuesta *no* debe haber sumandos que aparecen en la solución de la ecuación homogénea.

|  |  |
| --- | --- |
|  | propuesta |
| (*a* no es raíz de la ecuación característica) |  |
| (*a* es raíz de multiplicidad *t* de la ecuación característica) |  |
| Polinomio de grado *k* y 1 no es raíz de la ecuación característica | Polinomio genérico de grado *k* |
| Polinomio de grado *k* y 1 es raíz de multiplicidad *t* de la ecuación característica | Polinomio genérico de grado *k* multiplicado por |
| ó |  |

Caso especial 1:

1. Proponer una solución para
2. Proponer una solución para
3. La solución será .

Caso especial 2:

1. Proponer una solución para
2. Proponer una solución para
3. La solución será . Luego, comparar con la solución del homogéneo y arreglar si es necesario.

## Sucesiones importantes

|  |  |  |  |
| --- | --- | --- | --- |
| ***Interés*** | Fibonacci | Torres de Hanoi | Desarreglos |
| an = 1,12.an-1 | Fn = Fn-1 + Fn-2 | hn = 2hn-1 + 1 | dn = (n – 1).(dn-1 + dn-2) |

# **Unidad 4: Relaciones**

## Definiciones

Producto cartesiano:

Relación n-aria: dado un conjunto A se llama relación R en conjunto A R ⊆ A×A. Una relación se puede definir por extensión (mencionando todos sus elementos) o por comprensión (dando una característica de los elementos).

Relación ‘R’: Siendo x∈A, y∈A, decimos que xRy (x,y)∈R.

Relación inversa: dada , la relación inversa es tal que:

Repaso de funciones

Sean A y B dos conjuntos. Una relación es función si:

a ∈ A / f(a) = b0  f(a) = b1 (b0, b1 ∈ B b0 ≠ b1) (No existe elemento del dominio que tenga dos imágenes)

Sea función, a ∈ A, b ∈ B:

* f es inyectiva ⬄ a1 ≠ a2 ⇒ f(a1) ≠ f(a2) (Para puntos distintos del dominio, distintas imágenes)
* f es sobreyectiva ⬄ ∀ b ∈ B, ∃ a ∈ A / f(a) = b (La imagen de A es todo B)
* f es biyectiva ⬄ f es inyectiva y sobreyectiva (Si es biyectiva existe la inversa)

## Propiedades de las relaciones

Sea R una relación en el conjunto A.

1. R es *reflexiva* ∀ x ∈A: xRx
2. R es *simétrica* ∀ x,y ∈A : (xRy **→** yRx)
3. R es *transitiva* ∀ x,y,z ∈A : (xRy ∧ yRz) **→** xRz
4. R es *antisimétrica* ∀ x,y ∈A : (xRy ∧ yRx **→** x=y)

*Nota*: Todo elemento cumple las tres primeras consigo mismo. Cuidado con la 4º: no simétrica ≠ antisimétrica.

## Matriz de una relación

Sea R una relación en un conjunto *finito* A. La misma puede representarse matricialmente por:

siendo *n=|A|* definida por

Relación de orden entre matrices booleanas: . Es decir, una matriz C es menor a D si D tiene al menos los mismos 1 en las mismas posiciones que C.

Sea *I* la matriz identidad de *n x n*. Entonces:

* R es *reflexiva*
* R es *simétrica*
* R es *antisimétrica* (el producto se entiende posición por posición)
* R es *transitiva*

## Relaciones de equivalencia y de orden

|  |  |
| --- | --- |
| **Relación de equivalencia (~)** | **Relación de orden ()** |
| * Reflexividad * Simetría * Transitividad | * Reflexividad * Antisimetría * Transitividad |

* Orden total: ∀ x,y ∈ A : (x*R*y ∨ y*R*x). En el diagrama de Hasse se ve una línea recta.
* Orden parcial: x,y ∈ A : (x~~R~~y ∧ y~~R~~x)

(Si no es orden total, es orden parcial.)

Clase de equivalencia: sea R una *relación de equivalencia* en A. Se llama clase de equivalencia de un , al conjunto

**Teorema**: sea R una relación de equivalencia en A. Se verifica:

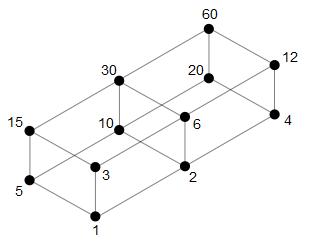
Conjunto cociente: . El conjunto cociente es una partición de A.

Partición: es una partición del conjunto A si y solo si:

Congruencia módulo n: En , y para , se define la relación

Diagrama de Hasse: representación gráfica simplificada de un conjunto (finito) ordenado parcialmente. Con ellos se eliminan los lazos de reflexividad y los atajos de transitividad. Si dos elementos están relacionados, digamos **aRb**, entonces dibujamos **b** a un nivel superior de **a**.

*Ejemplo*: sea el conjunto *A* = {1, 2, 3, 4, 5, 6, 10, 12, 15, 20, 30, 60} (todos los divisores de 60). Este conjunto está ordenado parcialmente por la relación de divisibilidad. Su diagrama de Hasse puede ser representado como sigue.



## Elementos particulares

Sea R una relación de orden en A:

Maximal: x0 es *maximal* de A x ∈ A : x0Rx (x0 no se relaciona con nadie).

Minimal*:* x0 es *minimal* de A x ∈ A : xRx0 (No hay elementos que se relacionen con el x0.)

Sea X un subconjunto de A:

Cota Superior: x0 ∈ A es *Cota Superior* de X ∀x ∈ X : xRx0.

Cota Inferior: x0 ∈ A es *Cota Inferior* de X ∀x ∈ X : x0Rx.

Supremo: s ∈ A es el *Supremo* de X s es la *menor* de todas los cotas superiores ∀x ∈ X : xRs.

Ínfimo: i ∈ A es *Ínfimo* de X i es la *mayor* de todas las cotas inferiores ∀x ∈ X : iRx.

Máximo: M ∈ A es *Máximo* de X M es supremo de X y M ∈ X.

##### Mínimo: m ∈ A es *Mínimo* de X m es ínfimo de X y m ∈ X.

# **Unidad 5: Álgebras de Boole**

## Definiciones y axiomas

Álgebra de Boole: Sea K () un conjunto no vacío que contiene dos elementos especiales, 0 (cero o elemento neutro) y 1 (uno o elemento unidad) sobre el cual definimos las operaciones cerradas +, • y el complemento. Entonces =(K, 0, 1, +, •, ) es un Álgebra de Boole si cumple las siguientes condiciones:

|  |  |
| --- | --- |
| A1) Axioma de conmutatividad | x + y = y + x  x.y = y.x |
| A2) Axioma de asociatividad | (x + y) + z = x + (y + z) = x + y + z  (x.y).z = x.(y.z) = x.y.z |
| A3) Axioma de la doble distributividad | x.(y + z) = x.y + x.z  x + (y.z) = (x + y).(x + z) |
| A4) Axioma de existencia de elementos neutros | x + 0 = x  x.1 = x |
| A5) Axioma de existencia de complementos | x + = 1  x. = 0 |

Expresión dual: se obtiene cambiando todos los +(•) por • (+) y los 0(1) por 1(0).

Principio de dualidad: en toda álgebra de Boole, si una expresión es válida, su expresión dual también lo es.

|  |  |
| --- | --- |
| 1) Ley del doble complemento: Observación: ∧ ≡ • ≡ ∩  ∨ ≡ + ≡ ∪ | = x |
| 2) Leyes de Morgan: | 1. = . 2. = + |
| 3) Leyes conmutativas: | 1. x + y = y + x 2. x.y = y.x |
| 4) Leyes asociativas: | 1. x + (y + z) = (x + y) + z 2. x.(y.z) = (x.y).z |
| 5) Leyes distributivas: | 1. x + (y.z) = (x + y).(x + z) 2. x.(y + z) = xy + xz |
| 6) Leyes de idempotencia: | 1. x + x = x 2. x.x = x |
| 7) Leyes de identidad: | 1. x + 0 = x 2. x.1 = x |
| 8) Leyes de inversos: | 1. x + = 1 2. x. = 0 |
| 9) Leyes de acotación: | 1. x + 1= 1 2. x.0 = 0 |
| 10) Leyes de absorción: | a) x + xy = x x + y = x + y  b) x.(x + y) = x x.( + y) = x.y |

|  |  |
| --- | --- |
| **Permitido** | **Prohibido** |
| * x + y = 0 → (x = 0) ∧ (y = 0) * x.y = 1 → (x = 1) ∧ (y = 1) * x + = z + ∧ x + y = z + y → x = z * x + = x.y → x = y | * x.y = 0 → (x = 0) (y = 0) * x + y = y + z → x = z |

## Funciones booleanas

Función booleana: . Dadas *n*variables, existen funciones booleanas posibles.

PROBLEMA

TABLA

EXPRESIÓN de *f*

EXPRESIÓN SIMPLIFICADA

CIRCUITO

|  |  |
| --- | --- |
| **MINITERMINOS** | **MAXITERMINOS** |
| m = x.y.z | M = x + y + z |
| Forma canónica, normal, normal disyuntiva SP: suma booleana de minitérminos. | Forma canónica, normal, normal conjuntiva PS: producto booleano de maxitérminos. |
| f(x,y,z) ≡ suma de los minitérminos que dan 1 | f(x,y,z) ≡ producto de los maxitérminos que dan 0 |
| Codificación: x **→** 1, **→** 0 | Codificación: x **→** 0, **→** 1 |

**Observación:**

La suma de los minitérminos de una función producto de los maxitérminos que no aparecen en la SP.

Σ m(0, 1, 3, 5, 7) = Π M(2, 4, 6)

Orden en un álgebra de Boole: sea = (K,+,,0,1,-) un álgebra de Boole. En K se define:

**Teorema**: . Todo álgebra de Boole está acotada.

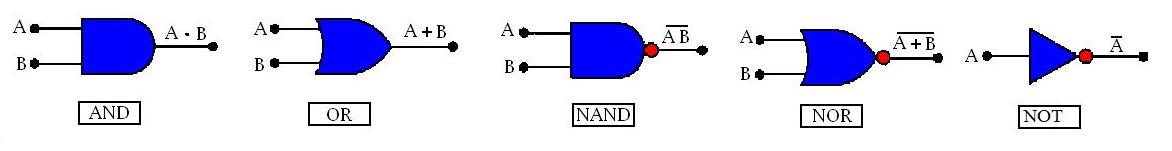
“0”

Átomo de un álgebra de Boole: es un *átomo* de B

∀y ∈ B: (y ≤ y = 0 y = )

*Nota*: Si B tiene *n* átomos ⇒ B tiene *2n* elementos.

Circuitos lógicos:



## Propiedades de los átomos

1. átomo (El producto de cualquier elemento de B con un átomo es 0 o es el átomo)
2. x0, x1 átomos distintos ⇒ x0.x1 = 0 (Si hay dos átomos distintos el producto entre ellos es 0)
3. Sean átomos de B (Si hay un x que multiplicado por cada uno de los átomos da 0, x es el 0)

**Teorema**: sean los átomos de B. Entonces tales que .

**Teorema**: , con átomo de B.

*Nota:* Si *n* es la cantidad de variables de f, el número máximo de términos es 2n.

## Mapa de Karnaugh

##### Para simplificar una función booleana. Se colorean los cuadrados de los minitérminos correspondientes y luego se escribe cada término, teniendo en cuenta que si un cuadrado tiene un vecino (abajo, arriba, derecha o izquierda) este último no se escribe.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **xy\zw** | **00** | **01** | **11** | **10** |
| **00** | 0 | 1 | 3 | 2 |
| **01** | 4 | 5 | 7 | 6 |
| **11** | 12 | 13 | 15 | 14 |
| **10** | 8 | 9 | 11 | 10 |

f = ∑ m(1, 3, 9, 11, 14, 6)

f = (w. + z..y) (simplificada)

## Isomorfismos entre álgebras de Boole

##### Isomorfismo entre dos álgebras de Boole: sean B1 = (K1, +1, •1, 01, 11, 1) y B2 = (K2, +2, •2, 02, 12, 2) dos álgebras de Boole. Se dice que B1 y B2 (#B1 = #B2) son isomorfos ∃ biyectiva tal que:

##### El número de isomorfismos posibles es (#B1)!

##### *Propiedades*:

1. f(01) = 02
2. f(11) = 12
3. f(átomo B1) = átomo B2
4. x R1 y **→** f(x) R2 f(y)

# **Unidad 6: Teoría de grafos**

## Definiciones de grafos y digrafos

Grafo no orientado: terna G = (V,A,δ) que representa una relación entre un conjunto finito de Vértices () y otro conjunto finito de Aristas (A), y δ es la función de incidencia.

δ: A **→** X(V), siendo X(V) = {X: X ⊆ V |X|= 1 o 2}.

Si δ(a) = {u,v} entonces

Grafo orientado / digrafo*:* terna D = {V,A,ϕ) con que representa una relación entre un conjunto finito de Vértices y otro conjunto finito de Aristas, y ϕ es la función de incidencia.

ϕ: A **→** V x V.

Si ϕ(a) = (v,w) entonces

## Aristas, vértices, caminos y grafos

**Aristas**

Aristas adyacentes: aristas que tienen un solo extremo en común.

Arista paralelas o múltiples: son aristas paralelas . Es decir, sii ϕ no es inyectiva.

Lazo o bucle: arista que une un vértice con sí mismo.

Arista incidente: Se dice que “*e* es incidente en *v*” si *v* esta en uno de los vértices de la arista *e*.

Extremo (para digrafos): Un extremo es inicial(final) si es el primer(ultimo) vértice de la arista.

Aristas paralelas (para digrafos): Si E.I(a) = E.I(b) ∧ E.F(a) = E.F(b) en otro caso son anti paralelas.

Puente: Es la arista que al sacarla el grafo deja de ser conexo.

##### Vértices

Vértices adyacentes: Se dice que “v y w son adyacentes” si existe una arista entre los dos vértices.

* Un vértice es adyacente a sí mismo si tiene lazo.

Grado de un vértice: gr(v) es la cantidad de aristas que inciden en él. Los lazos cuentan doble.

* Se dice que un vértice es ‘par’ o ‘impar’ según lo sea su grado.
* La cantidad de vértices de grado impar es un número par.
* Si gr(v) = 0, v es un vértice aislado.

Grado positvo (para digrafos): es la cantidad de veces que se usa el vértice como extremo final.

Grado negativo (para digrafos): es la cantidad de veces que se usa el vértice como extremo inicial.

* grtotal(v)=
* grneto(v)=
* El lazo cuenta como arista incidente positiva y negativamente en el vértice.

Vértice de aristas múltiples: Es aquel que tiene más de un arista.

##### Caminos

Camino: sucesión finita no vacía de aristas distintas que contengan a vx y vy en su primer y último término. Así: {vx,v1},{v2,v3},...,{vn,vy}

Longitud del camino: número de aristas de un camino.

Circuito o camino cerrado: camino en el cual .

Camino simple: camino que no repite vértices.

Circuito simple: circuito que no repite vértices salvo el primer y último vértice.

Ciclo: circuito simple que no repite aristas.

* Circuito simple de longitud ≥ 3 en grafos (≥ 2 en digrafos) es un ciclo.

*Nota:* Si ∀v ∈ V gr(v) ≥ 2 ⇒ el grafo tiene un circuito.

##### Grafos

Orden de un grafo: Es su número de vértices.

Grafo acíclico: grafo que no tiene ciclos.

Grafo conexo: grafo tal que dados 2 vértices distintos es posible encontrar un camino entre ellos. camino de a )

Grafo simple: grafo que carece de aristas paralelas y lazos.

Grafo regular: Aquel con el mismo grado en todos los vértices.

Grafo k-regular: G=(V,A,) es k-regular

Grafo bipartito: Es aquel con cuyos vértices pueden formarse dos conjuntos disjuntos de modo que no haya adyacencias entre vértices pertenecientes al mismo conjunto.

v3

v1

v4

v2

v5

Grafo Kn,m: grafo bipartito simple con la mayor cantidad de aristas.

* #= n.m

Grafo Kn: grafo simple con *n* vértices y la mayor cantidad de aristas.

* #=

Grafo completo: grafo simple con mayor cantidad de aristas. Todos están conectados con todos.

* ∀v ∈ V, gr(v) = #V – 1.
* Si G(V,A) es completo ⇒ G es regular (No vale la recíproca)
* Dos grafos completos con mismo #V son isomorfos.

Grafo complemento: dado G=(VG,AG) simple se llama grafo complemento a tal que . Es el grafo G’ que tiene conectados los vértices no conectados de G y desconectados los vértices conectados de G.

v5

v3

v2

v4

G

v2

G’

v3

v1

v1

v5

v4

* G ∪ G’ = Grafo completo.
* Si dos grafos son complementarios, sus isomorfos también.
* Sea

Grafo plano: Aquel que admite una representación bidimensional sin que se crucen sus aristas.

Grafo ponderado: Es el grafo en cual cada arista tiene asignado un n° real positivo llamado peso.

Digrafo: Grafo con todas sus aristas dirigidas. Por tanto, los pares de vértices que definen las aristas, son pares ordenados.

Digrafo conexo: Si su grafo asociado es conexo.

Digrafo fuertemente conexo: ∀v ∈ V ∃ camino que me permite llegar a cualquier otro vértice.

Digrafo k-regular: D=(V,A,) es k-regular

Subgrafo de G: Dado G = (V, A), G’ = (V’, A’) es subgrafo de G si V’⊆ V y A’ ⊆ A

Grafo parcial de G: Dado G = (V, A), G’ = (V’, A’) es grafo parcial de G si V’⊆ V y A’ ⊆ A

Multigrafo: Grafo que tiene alguna arista múltiple.

* Un multigrafo se transforma en grafo añadiendo un vértice en mitad de cada arista múltiple.

Pseudografo: Grafo con algún lazo.

## Grafos de Euler

Grafo de Euler: grafo en el cual se puede encontrar un ciclo o un camino de Euler.

* Camino de Euler:camino que no repite aristas.
* Circuito de Euler: circuito que no repite aristas.

**Teorema de Euler:**

* Para grafos conexos:
* G tiene un Camino de Euler ⬄ G tiene exactamente 2 vértices de grado impar.
* G tiene un Circuito de Euler ⬄ G tiene exactamente 0 vértices de grado impar.
* Para digrafos:
* G tiene un Camino de Euler ⬄ ∃ u,w ∈ V (u ≠ w)
* G tiene un Circuito de Euler ⬄ ∀v ∈ V

Grafo de Hamilton: grafo en el cual es posible hallar un camino o circuito de Hamilton.

* Camino de Hamilton: Es un camino que no repite vértices. (Puede no pasar por todas las aristas)
* Circuito de Hamilton: Es un circuito que no repite vértices. (Puede no pasar por todas las aristas)

**Teorema de Ore:** Si un grafo es conexo con y ⇒ G es Grafo Hamiltoniano.

**Teorema de Dirac:** un grafo *simple* con es Hamiltoniano si

1. Isomorfismos de grafos

Dados G=(V, A) y G’=(V’,A’), se denomina *isomorfismo de G a G’* a la aplicación biyectiva f tal que para a,b ∈ V, {a,b} ∈ A ⇔ se cumple {f(a),f(b)} ∈ A’. Es decir, la aplicación que relaciona biyectivamente pares de vértices de A con pares de vértices de A’, de modo que los vértices conectados siguen estándolo.

* #V = #V’ y #A = #A’
* Se cumple que δ(a)=δ(f(a))
* Si dos grafos son isomorfos, sus complementarios también.
* G y G’ tienen igual cantidad de vértices aislados.
* G y G’ tienen igual cantidad de lazos o bucles.
* Se mantienen los caminos.
* Se mantienen los ciclos.
* Si dos grafos complementarios son isomorfos se los llama auto complementarios.
* Dos grafos simples G1 y G2 son isomorfos ⇔ para cierto orden de sus vértices las MA son iguales.

Automorfismo: Es un isomorfismo en sí mismo. f(a) = a.

## Representación de grafos por matrices

|  |  |  |
| --- | --- | --- |
|  | ***Grafos***  a3  v3  v4  a4  a1  v1  v2  v5  a6  a5  a2 | ***Digrafos***  v4  v2  a4  a6  a5  a3  a1  v1  v3  a2 |
| **Matriz de adyacencia** | () tal que: con cantidad de aristas con extremos y   * Matriz simétrica. * gr(vi) = Σaij + 2.aii (i ≠ j)  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  | **v1** | **v2** | **v3** | **v4** | **v5** | | **v1** | 0 | 1 | 1 | 0 | 0 | | **v2** | 1 | 1 | 0 | 1 | 0 | | **v3** | 1 | 0 | 0 | 2 | 0 | | **v4** | 0 | 1 | 2 | 0 | 0 | | **v5** | 0 | 0 | 0 | 0 | 0 |   gr(v1)  gr(v1) | () tal que concantidad de aristas con E.I en vi y E.F en vj  * No necesariamente simétrica.  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  | **v1** | **v2** | **v3** | **v4** | **v5** | | **v1** | 0 | 0 | 1 | 0 | 0 | | **v2** | 1 | 1 | 0 | 0 | 0 | | **v3** | 0 | 0 | 0 | 1 | 0 | | **v4** | 0 | 1 | 1 | 0 | 0 | | **v5** | 0 | 0 | 0 | 0 | 0 | |
| **Matriz de incidencia**  | |  0   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | |  | **a1** | **a2** | **a3** | **a4** | **a5** | **a6** | | **v1** | 1 | 0 | 0 | 0 | 0 | -1 | | **v2** | -1 | ±1 | 1 | 0 | 0 | 0 | | **v3** | 0 | 0 | 0 | -1 | -1 | 1 | | **v4** | 0 | 0 | -1 | 1 | 1 | 0 | | **v5** | 0 | 0 | 0 | 0 | 0 | 0 | | () tal que , con   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | |  | **a1** | **a2** | **a3** | **a4** | **a5** | **a6** | | **v1** | 1 | 0 | 0 | 0 | 0 | 1 | | **v2** | 1 | 2 | 1 | 0 | 0 | 0 | | **v3** | 0 | 0 | 0 | 1 | 1 | 1 | | **v4** | 0 | 0 | 1 | 1 | 1 | 0 | | **v5** | 0 | 0 | 0 | 0 | 0 | 0 |   gr(v1)  | |  2 | () tal que  , con  gr+(v1)=Σaij,(aij>0)  gr -(v1)=Σaij,(aij<0) |

*Propiedad:* en la matriz , cada coeficiente indica la cantidad de caminos de longitud *k* que hay entre y .

Matriz de conexión: Dados G=(V,A,) con y . Se define la siguiente relación: .

Matriz de adyacencia booleana: sea un grafo G=(V,A,) con y . Se define la matriz de adyacencia de G a una matriz booleana de tal que:

Matriz de incidencia booleana: sea un grafo G=(V,A,) con y . Se define la matriz de adyacencia de G a una matriz booleana de tal que:

## Niveles

##### Vértice alcanzable: sea D=(V,A) un digrafo. Se dice que se alcanza de camino dirigido de a .

##### Niveles de un digrafo: Un conjunto vértices N constituye o está en nivel superior a otro conjunto de vértices K si ningún vértice de N es alcanzable desde algún vértice de K.

Dibujar MA

i = 1

while MA:

Nivel i = vi’s tales que sus filas y columnas en MA sean nulas

MA = MA – {columnas y filas que sean nulas}

i = i + 1

C

D

F

B

G

A

E

A

B

C

D

E

F

G

Nivel 1: A,G

Nivel 2: B

Solo flechas descendentes!

Nivel 3: E

Nivel 4: C

Nivel 5: F

Nivel 6: D

## Algoritmos de camino mínimo

Objetivo: Hallar el camino mínimo de S a L:

* λ(v) es la etiqueta del vértice v.
* i es un contador.

**Algoritmo de Moore o BFS (*Breadth First Search*)**

* Dado un grafo o digrafo no ponderado, calcula la distancia entre dos vértices.

λ(S) = 0

i = 0

while (vértices adyacentes a los etiquetados con i no etiquetados):

λ(v) = i+1

if (L is etiquetado): break

i = i+1

##### Algoritmo de Dijkstra

* Dado un grafo o digrafo con pesos no negativos, calcula caminos mínimos del vértice a todos los vértices.

λ(S) = 0

for v in V:

λ(v) = ∞

T = V

while (L T):

Elijo v ∈ T con mínimo λ(v) adyacente al último etiquetado

∀x / x adyacente v:

λ(x) = min{λ(x), λ(v) + a(v,x)}

T = T – {v}

##### Algoritmo de Ford

* Solo para digrafos, acepta pesos negativos y detecta circuitos negativos.

λ(S) = 0

for v in V:

λ(v) = ∞

j = 1

while ( j ≠ |V|):

T ={v ∈ V / v sea adyacente al último etiquetado}

∀x ∈ V, ∀v ∈ T :

λ(v) = min{λ(x), λ(v) + a(v,x)}

Si no hubo cambios: break

Else: j = j + 1

return T

# **Unidad 7: Árboles**

## Definiciones

Árbol: G=(V,A) es un árbol ⇔ ∀ u,v ∈ V ( ∨ ∃! camino simple de u a v)

**Teorema 1**: dado un grafo G=(V,A). Las siguientes afirmaciones son equivalentes:

1. G es conexo y acíclico
2. G es acíclico y si se le agrega una artista deja de serlo
3. G es conexo y si se le elimina una arista deja de serlo
4. G es árbol

Teorema 2: dado un grafo G=(V,A). Las siguientes afirmaciones son equivalentes:

1. G es conexo y acíclico
2. G es conexo y
3. G es acíclico y

Propiedad: si G es un árbol con ⇒ hay al menos 2 vértices de grado 1.

Bosque: un grafo G=(V,A) es bosque ⇔ G es acíclico.

* Los bosques son grafos no conexos cuyas componentes conexas son árboles.
* , siendo *t* la cantidad de árboles del bosque.

##### Arboles con raíz: G=(V,A) digrafo conexo es un árbol con raíz ⇔

Hoja / terminal: Vértice sin hijos.

Vértice interno: Vértice con hijos.

Árbol n-ario: todos los nodos tienen a lo sumo *n* hijos.

Árbol n-ario completo: todos los nodos tienen 0 o *n* hijos.

Nivel de un vértice: número de aristas que le separan de la raíz. La raíz tiene nivel 0.

Altura de un árbol*:* máximo nivel de sus vértices.

Árbol equilibrado: las hojas llegan al mismo nivel.

**Teorema:** Si T = (V, A) es una árbol m-ario completo con *i* vértices internos entonces:

## Árboles generadores

Árbol generador: T=(,) es un árbol generador de G=(,) ⇔

Árbol generador minimal: es un árbol generador, de peso mínimo. No es único.

**Teorema:** G es un grafo no dirigido y conexo ⇔ G tiene árbol generador.

## Algoritmos para hallar un árbol generador mínimo

Sea G = (V, A) un grafo conexo ponderado. Existen dos algoritmos para hallar un árbol generador mínimo de G.

##### Algoritmo de Prim

v = vértice cualquiera de G

T = {v}

while (|T| ≠ |V|):

a = arista de mínimo peso incidente en un v ∈ T y un w ∉ T

T = T + {w}

return T

##### Algoritmo de Kruskal

a = arista de mínimo peso de G

T = {a}

while (|T| < |V|-1):

b = arista de mínimo peso tal que b ∉ T y T + {b} es acíclico

T = T + {b}

return T

# **Unidad 8: Redes de transporte**

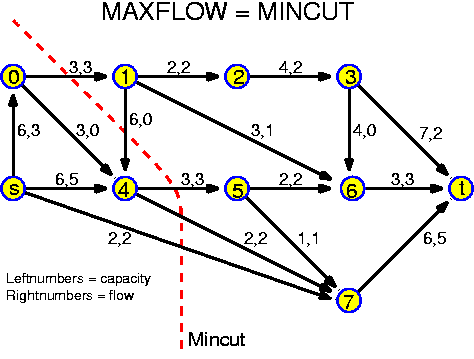
## Definiciones

Red de transporte: sea G = (V, A) un digrafo conexo y sin lazos. G es una red de transporte si se verifican:

1. *Vértice Fuente*: ∃! vértice *f* ∈ V / (no llegan flechas)
2. *Vértice Sumidero*: ∃! vértice *s* ∈ V / (no salen fleches)
3. *Capacidad de la Arista*: ∃ una función / si a = (vi, vj) ∈ A, C(a) = Cij

##### Flujo de una red: Si G = (V, A) es una red de transporte se llama *flujo de G* a una función F: A → N0 tal que:

1. ∀a ∈ A: F(a) ≤ C(a) (Si F(a) = C(a) se dice que la arista está *saturada*)
2. ∀v ∈ V (v ≠ *f* , v ≠ *s)* se tiene que (Flujo entrante = Flujo saliente)

****

**Teorema 1:** Si F es el flujo asociado a una red de transporte se cumple que

(Todo lo que sale de la fuente llega al sumidero)

##### Valor del flujo: suma de los flujos de todas las aristas que salen del vértice fuente:

Corte de una red: Un corte (P, ) en una red de transporte G = (V, A) es un conjunto P tal que:

Capacidad de un corte: Se llama capacidad de un corte (P,) al número: Es la suma de todas las aristas incidentes en v y w tal que v ∈ P y w ∈ . (Las aristas por donde pasa el corte).

**Teorema 2**: Sea F un flujo de la red G = (V, A) y sea (P, ) un corte de G. Entonces: C(P, ) ≥ val(F)

##### Teorema 3 (del flujo Máximo y Corte Minimal): Si C(P,) = val(F) ⇒ el flujo es máximo y el corte es minimal.

F = 0

saturada

P

**Teorema 4**: C(P, ) = val(F) ⇔

## Algoritmo de Ford-Foulkerson

Se utiliza para hallar el flujo máximo en una red de transporte.

Dada una red de transporte G = (V, A), con *f* (fuente) y *s* (sumidero):

* λ(v) función de etiquetación de v.
* ek capacidad residual de vk.

1. Poner en la red un flujo compatible.
2. Etiqueto la fuente con
3. Para cualquier vértice *x* adyacente a *a*, etiquetamos a *x*:
4. Si , etiquetamos *x* con ).
5. Si , no etiquetamos *x.*
6. Mientras exista (*x a*) en V tal que *x* esté etiquetado y exista una arista (x,y) tal que y no esté etiquetado, etiquetamos a y:
7. Si , etiquetamos *y* como
8. Si , no etiquetamos *y*.
9. Mientras exista (*x a*) en V tal que *x* esté etiquetado y exista una arista (x,y) tal que y no esté etiquetado, etiquetamos a y:
10. Si , etiquetamos *y* como
11. Si , no etiquetamos *y*.