Curso esquemático de Matemática Discreta

Beatriz Graña Otero

21 de septiembre de 2010

Índice general

Capítulo 1

Relaciones entre conjuntos.

1.1. Relaciones.

Definición 1. Sea X un conjunto. Una relación en X es un subconjunto R del producto cartesiano $X \times \overset{n)}{\ldots} \times X$ ($R \subset X \times \overset{n)}{\ldots} \times X$).

Definición 2. Sea X un conjunto. Una relación binaria en X es un subconjunto R del producto cartesiano $X \times X$ ($R \subset X \times X$).

En lo que sigue se dirá relación para entender que es una relación binaria.

Definición 3. Sea X un conjunto y $R \subset X \times X$ una relación en X, se dice que $x \in X$ se relaciona con $y \in X$ y se denota por $x\ R\ y$ si $(x, y) \in R$.

Definición 4. Sea X un conjunto y $R \subset X \times X$ un relación en X.

1. Se dice que R es *reflexiva* si $(x, x) \in R$ para todo $x \in X$.
2. Se dice que R es *simétrica* si $\forall\ (x, y) \in R$ se verifica que $(y, x) \in R$.
3. Se dice que R es *antisimétrica* si cuando $(x, y) \in R$ e $(y, x) \in R$ entonces $x = y$.
4. Se dice que R es *transitiva* si $\forall\ (x, y) \in R$ y $\forall\ (y, z) \in R$ se tiene que $(x, z) \in R$.

Definición 5. Sea X un conjunto y $R \subset X \times X$ una relación en X.

1. Se dice que R es una ***relación de equivalencia*** si R verifica las propiedades reflexiva, simétrica y transitiva.
2. Se dice que R es ***una relación de orden*** si R verifica las propiedades reflexiva, antisimétrica y transitiva.

1.1.1. Representación de relaciones binarias

Puesto que una relación binaria en un conjunto X es un subconjunto de $R \subset X \times X$, ésta se puede ver como los puntos del plano cartesiano cuyos ejes son el conjunto X. Es decir, mediante una tabla.

La propiedad reflexiva implica que el conjunto $\Delta = \{(x, x) \mid x \in X\} \subset R$ es un subconjunto de la relación.

La propiedad simétrica se refleja en la gráfica haciéndola simétrica respecto de Δ.

La propiedad antisimétrica se ve comprobando que no existen puntos pertenecientes a la tabla simétricos respecto de dicha diagonal.

Para observar si R disfruta de la propiedad transitiva necesitamos hacer uso de la representación matricial de las relaciones binarias.

Definición 6 (matriz de adyacencias)**.** Sea $X = \{x_1, \ldots, x_n\}$ un conjunto finito y $R \subset X \times X$ una relación en X. Se define ***la matriz de adyacencias*** asociada a la relación R como la matriz $M = (m_{ij})$, donde $m_{ij} = 1$ si $x_i \; R \; x_j$ y cero en caso contrario.

Si la relación R verifica la propiedad reflexiva, la matriz M de adyacencias tendrá en la diagonal principal todos unos. Si es una relación simétrica, la matriz será simétrica.

Si R es una relación antisimétrica, la matriz de adyacencias cumple que si $m_{ij} = 1$ entonces $m_{ji} = 0$.

Para la propiedad transitiva se verifica la siguiente proposición:

Proposición 1. *Sea $X = \{x_1, \ldots, x_n\}$ un conjunto finito, $R \subset X \times X$ una relación en X, M la matriz de adyacencias de R y $M.M = (n_{ij})$ la matriz producto de M por si misma. Entonces R es transitiva si y sólo si la matriz M verifica que $n_{ij} \neq 0$ implica $m_{ij} \neq 0$.*

Demostración. Supongamos que R es transitiva y sea $n_{ij} \neq 0$ (recuerda que el $n_{ij} = \sum_l m_{il} m_{lj}$), entonces existe un elemento $x_k \in X$ tal que $x_i \; R \; x_k$ y $x_k \; R \; x_j$ por la propia definición de M^2. Ya que R es transitiva, se concluye que $x_i R x_j$ y por tanto $m_{ij} \neq 0$.

Recíprocamente, supongamos que es cierto que si $n_{ij} \neq 0$ entonces $m_{ij} \neq 0$ y tomemos x_i, x_j y $x_k \in X$, con $i \neq j \neq k$ tal que $x_i \; R \; x_k$ y $x_k \; R \; x_j$. Estas condiciones significan que $n_{ij} \neq 0$ y por hipótesis ello implica que $m_{ij} \neq 0$. Pero la definición de matriz de adyacencias esta última condición es que $x_i \; R \; x_j$, y se demuestra la transitividad de R. □

Otra forma de representar relaciones binarias de conjuntos finitos es mediante un grafo dirigido que se tratará con profundidad en el tema siguiente, pero se puede anticipar que los elementos serán los vértices del grafo y los elementos de la relación serán las aristas que unen los vértices cuya dirección dice que el punto origen está relacionado con el punto final de la arista.

1.2. Relaciones de equivalencia.

Definición 7. Sea X un conjunto y $R \subset X \times X$ una relación de equivalencia en X. Sea $x \in X$, se define la clase de equivalencia de x por la relación R al conjunto

$$[x] = \{y \in X \mid yRx\}$$

.

Nota 1. Cuando en un conjunto cualquiera existe un relación de equivalencia, se suele denotar por $\sim$ y las clases de un elemento x del conjunto respecto de ella por $[x]$ o por $\overline{x}$ indistintamente.

Proposición 2 (Propiedades)**.** *Sea X un conjunto y $R \subset X \times X$ una relación de equivalencia en X.*

1. *Para todo $x \in X$, la clase $[x] \neq \emptyset$ de equivalencia es distinta del conjunto vacío.*
2. *xRy si y sólo si $[x] = [y]$.*
3. *$[x] \neq [y]$ si y sólo si $[x] \cap [y] = \emptyset$.*
4. *El conjunto $X = \sqcup_{x \in X} [x]$ es unión disjunta de sus clases de equivalencia.*

Definición 8. Sea X un conjunto y $R \subset X \times X$ una relación de equivalencia en X. Se define el conjunto cociente X/R por la relación R al conjunto

$$X/R = \{[x] \mid x \in X\}$$

1.3. Relaciones de orden.

Definición 9. Sea X un conjunto y $R \subset X \times X$ una relación de orden en X, se dice que dos elementos $x, y \in X$ son ***comparables*** si $(x, y) \in R$ ó $(y, x) \in R$. Si todos los elementos de R son comparables se dice la relación de orden R es de orden total y que el conjunto X está totalmente ordenado por R. En otro caso se dirá que la relación de orden R es de orden parcial y que el conjunto X está parcialmente ordenado por R

Nota 2. Cuando en un conjunto cualquiera existe un relación de orden, es frecuente ver que la relación se denota por $\leq$ por similitud con la relación binaria de orden usual de los números reales.

Definición 10 (Elementos característicos de R)**.** Sea X un conjunto y R una relación de orden en X.

1. Se dice que $x \in X$ es un elemento ***maximal*** (análg. ***minimal***) de X si se verifica que xRz (análg. zRx) implica que $x = z$.

2. Sea $Y \subset X$, se dice que un elemento $x \in X$ es un ***supremo*** (análg. ***ínfimo***) de Y si yRx (análg. xRy) para todo $y \in Y$ y además si existe $z \in X$ con la misma propiedad que disfruta x; es decir, yRz para todo $y \in Y$, entonces xRz (análg. zRx).

3. Se dice que $x \in X$ es un ***máximo*** (análg. ***mínimo***) de X si yRx para todo $y \in X$.

Las relaciones de orden poseen una forma particular de representación que es el *diagrama de Hasse*, cuando el conjunto es finito. Ésta es la forma más reducida de dibujarlo como grafo. Las propiedades reflexiva y transitiva no se dibujan, lo que significa que cada vértice se da por supuesto que está relacionado con él mismo y que si (a, b) y (b, c) son aristas del grafo, también lo es (a, c) aunque no se dibuje. Por último, se elige un distribución vertical en lugar de usar aristas dirigidas (flechas) en el sentido de que si a está por debajo de b unido por una arista, quiere decir que aRb.

Lema 1. *Sea (X, R) un conjunto finito parcialmente ordenado por R, entonces existe un elemento minimal.*

Demostración. Sea $x_1 \in X$ un elemento cualquiera del conjunto X. Si x_1 es un elemento minimal hemos terminado y se verifica el lema, en caso contrario existe un elemento $x_2 \in X$ tal que $x_2 R x_1$ con $x_2 \neq x_1$. De nuevo, si x_2 es minimal hemos terminado y en caso contrario existe $x_3 \in X$ tal que $x_3 R x_2$ y $x_3 \neq x_2$. Como el conjunto es finito, este proceso terminará dando un elemento minimal. □

Algoritmo de construcción de un orden total T que contenga a un orden parcial dado R

Partimos de X y elegimos un elemento minimal x_1. En el paso siguiente se considera el conjunto $X - \{x_1\}$ y la relación de orden parcial inducida en él. Se repite el proceso eligiendo un nuevo elemento minimal x_2. Se sigue el proceso hasta que no queden elementos en X y el orden total es el dado por

$$x_1 \; T \; x_2 \; T \; \cdots \; T \; x_n.$$

.

1.4. Clausura reflexiva, simétrica y transitiva.

Se puede presentar el problema de encontrar, dado un conjunto X y una relación R en él, la mínima relación con ciertas condiciones que contiene a R (en el sentido del menor subconjunto de $X \times X$ que contiene a R).

Definición 11. Sea X un conjunto finito y R una relación en él. Se define la **clausura transitiva** (resp. simétrica, reflexiva) al mínimo conjunto $C_t \subset X \times X$ (resp. C_s, resp. C_r) tal que $R \subset C_t$ (resp. $R \subset C_s$, $R \subset C_r$) y la relación que define C_t (resp. C_s, C_r) sea transitiva (resp. simétrica, resp. reflexiva).

reflexiva $C_r = R \cup \Delta(X)$, donde $\Delta(X) = \{(x, x) \mid x \in X\}$. Y la matriz de adyacencias de la nueva relación tiene en la diagonal principal todos unos.

simétrica $C_s = R \cup S(X)$, donde $S(X) = \{(y, x) \mid (x, y) \subset R\}$. Y la matriz de adyacencias de la nueva relación es simétrica y se obtiene sumándole a la matriz de adyacencia M de R la matriz M^t.

transitiva La clausura transitiva tiene un cómputo un poco más complicado y hay que tener en cuenta los siguiente lemas.

Lema 2. *Sea X un conjunto y R una relación en él. Si $x, y \in X$ y existen $x_1, \ldots, x_n \in X$ tales que los pares $(x, x_1), \ldots, (x_n, y) \in R$, entonces el par (x, y) pertenece a la clausura transitiva.*

Demostración. La demostración se hace por inducción sobre el número n igual al número de puntos que se necesitan para ir de x a y. Si $n = 0$, entonces $(x, y) \in R \subset C_t$ y no hay nada que demostrar. Si el $n = 1$, $(x, x_1) \in R$ y $(x_1, y) \in R$, luego $(x, y) \in C_t$.

Supongamos que es cierto el lema cuando son necesarios los $n - 1$ puntos $x_1, \ldots, x_{n-1}$ de X para salir de x y llegar a y. Por hipótesis de inducción, $(x, x_i) \in C_t$ y $(x_{i-1}, x_i) \in C_t$ para todo $1 \leq i \leq n$, pero como además $(x_n, y) \in R$ se puede concluir que $(x, y) \in C_t$. □

Lema 3. *Sea X un conjunto y R una relación en él. Si la relación R verifica que para $x, y \in X$, existen $x_1, \ldots, x_n \in X$ tales que los pares $(x, x_1), \ldots, (x_n, y) \in R$, entonces el par $(x, y) \in R$, se verifica que R es transitiva.*

Demostración. En particular, para $n = 1$, si $x, y, x_1 \in X$ y $(x, x_1) \in R$ y $(x_1, y) \in R$, luego $(x, y) \in R$ y la relación es transitiva.

□

Por tanto la clausura transitiva de una relación R en un conjunto X es:

$$T = \{(x, y) \in X \times X \mid \text{existe “un camino” que une } \ a \text{ y } b\}$$

y donde por camino se entiende que existen elementos $x_1, \ldots, x_n$ del conjunto X tales que

$$(x, x_1) \in R, (x_1, x_2) \in R, \ldots, (x_n, y) \in R.$$

Algoritmo para la clausura transitiva.

Sea X un conjunto y R una relación definida en él. Sea M una matriz de adyacencias asociada a R. Para calcular la clausura transitiva se procede como sigue:

- **Entrada** Matriz M (de dimensión $n \times n$ y n el número de elementos de X)
- Para $i = 2, \cdots, n$, $M := M + M^i$
- **Salida** M_{C_t} es la matriz que tiene un 1 donde M tiene una entrada no nula y un 0 donde M tiene una entrada nula y es la matriz de una relación C_t que es transitiva y contiene a R.

Capítulo 2

Teoría de Grafos

2.1. Grafos simples, multigrafos y digrafos.

Los grafos son una reunión finita de vértices y aristas que unen estos. Dependiendo de cómo son las aristas hay distintos tipos de grafos.

Definición 12. Un par $G = (V, A)$ es un *grafo simple* si V es un conjunto finito de **vértices** y A un conjunto de pares no ordenados de vértices distintos llamados **aristas no dirigidos**.

$$A \subset S^2V - \Delta(V) = \{\text{parejas no ordenadas de elementos distintos de } V\}$$

Definición 13. Un par $G = (V, A)$ es un *grafo dirigido simple* si V es un conjunto finito de vértices y A, las aristas dirigidos, un conjunto de pares ordenados de vértices distintos.

$$A \subset V \times V - \{(a, a) \mid a \in V\}$$

Como se observa de la definiciones, no se está permitiendo que estos grafos tengan varias aristas uniendo los mismo vértices (multigrafo) ni aristas que salgan y lleguen al mismo vértice

Definición 14. Un par $G = (V, A)$ es un *multigrafo* si $V = \{a_1, \ldots, a_n\}$ es un conjunto finito de vértices y A es una familia finita de pares no ordenados de vértices.

$$A \subset \{\{a_i, a_j\} \mid i, j \in \{1, \ldots, n\}\}$$

Definición 15. Sea $G = (V, A)$ un multigrafo, a las aristas de la forma $\{a, a\}$ con $a \in V$ se les denomina **lazos**.

Nótese que la diferencia entre un multigrafo y un grafo simple es que el segundo no posee lazos además el hecho de que las aristas formen un conjunto y no solo un familia quiere decir que no existen repeticiones y así no hay varias aristas con los mismos vértices.

Definición 16. Un par $G = (V, A)$ es un *multidigrafo* si $V = \{a_1, \ldots, a_n\}$ es un conjunto finito de vértices y A es una familia finita de pares ordenados de vértices.

$$A \subset \{(a_i, a_j) \mid i, j \in \{1, \ldots, n\}\}$$

En general, cuando se habla de que $G = (V, A)$ es un grafo, se considera en sentido amplio y puede tanto ser dirigido o no, con lazos o sin lazos y con aristas múltiples o no.

Definición 17. Sea $G = (V, A)$ un grafo.

1. Se dice que dos vértices $a, b \in V$ son *adyacentes* si $\{a, b\} \in A$ existe una arista que los une. Se dice que la arista *conecta* a a y a b, que *incide* con ellos y que estos son los *extremos* de la arista $\{a, b\}$.

2. Se denomina *grado* de $a \in V$ ($gr(a)$) al número de aristas que inciden en a. (Un lazo aporta dos unidades al grado de un vértice). Si un vértice tiene grado cero se dice que es un *vértice aislado.*

3. Si $G = (V, A)$ es un grafo dirigido y $(a, b) \in A$ una arista de él, se dice que a es el *vértice inicial* y b el *vértice final* de dicha arista. Se habla del *grado de entrada* ($gr^-(a)$) del vértice a y es el número de aristas que tienen a a como vértice inicial y del *grado de salida* ($gr^+(a)$) del vértice a y es el número de aristas que tiene a a como vértice final.

Teorema 1. *Sea $G = (V, A)$ un grafo no dirigido, entonces*

$$\Sigma_{v \in V} gr(v) = 2|A|$$

Demostración. Cada arista aporta dos unidades a la suma de los grados de los vértices de G, puesto que es incidente exactamente con dos vértices. □

Corolario 1. *Todo grafo $G = (V, A)$ tiene un número par de vértices de grado impar.*

Demostración. Se tiene que $2|A| = \Sigma_{v \in V} gr(v) = \Sigma_{v \in V_1} gr(v) + \Sigma_{v \in V_2} gr(v)$, donde V_1 es el conjunto de vértices de G de grado par y V_2 es el conjunto de vértices de grado impar. □

Nota 3. Un lazo aporta dos unidades al grado del vértice sobre el que está, si el grafo es dirigido, una de salida y una de entrada.

Teorema 2. *Sea $G = (V, A)$ un grafo dirigido, entonces*

$$|A| = \Sigma_{v \in V} gr^+(v) = \Sigma_{v \in V} gr^-(v)$$

2.1.1. Representación de grafos.

Ya se introdujo en las relaciones cómo éstas podían ser representadas mediante grafos. La manera más común de dibujar grafos es mediante puntos y segmentos, dirigidos o no, que en el caso de las relaciones los puntos son los elementos del conjunto y las aristas las relaciones que existen entre ellos.

Dependiendo de las necesidades de cada caso, los grafos pueden se dados mediante matrices de la siguiente manera.

Definición 18. Sea $G = (V = \{v_1, \ldots, v_n\}, A)$ un grafo simple, se llama **matriz de adyacencias** del grafo G a la matriz $M = (a_{ij}) \in Mat_{n \times n}(\{0, 1\}$, donde $a_{ij} = 1$ si existe una arista que une el vértice v_i al vértice v_j; es decir si $\{v_i, v_j\} \in A$. Y a_{ij} es cero en caso contrario.

Definición 19. Sea $G = (V = \{v_1, \ldots, v_n\}, A)$ un grafo cualquiera, se llama **matriz de adyacencias** del grafo G a la matriz $M = (a_{ij}) \in Mat_{n \times n}(\mathbb{Z})$, donde a_{ij} = número de aristas que conectan el vértice v_i con el vértice v_j. Si es dirigido, se pondrá el número de aristas que salen de v_i y llegan a v_j.

2.2. Isomorfismo y conexión de grafos.

2.2.1. Algunas definiciones.

Sea $G = (V, A)$ un grafo con V un conjunto finito.

Definición 20. Se dice que entre los vértices $v \in V$ y $v' \in V$ existe un **camino de longitud** n en G, si los vértices $v_1, \ldots, v_{n-1} \in V$ verifican que:

- Si G es dirigido, las aristas $(v, v_1), \ldots, (v_{n-1}, v') \in A$ son del grafo.
- Si G no es dirigido, las aristas $\{v, v_1\}, \ldots, \{v_{n-1}, v'\} \in A$ son del grafo.

Definición 21. Se dice que un camino en G es un **camino simple** si no se repiten los vértices por los que pasa.

Definición 22. Se dice que un camino simple en G es un **ciclo** si es un camino que sale y llega al mismo vértice.

Definición 23. Se dice que un camino en G es un **recorrido** si no se repiten las arista por las que pasa.

Definición 24. Se dice que un camino en G es un **circuito** si es un recorrido que empieza y termina en el mismo vértice.

Definición 25. Sea $G = (V, A)$ un grafo no dirigido, se dice que G es el **grafo completo** K_n de n vértices si $\forall\, v \in V$ es adyacente con todo el resto de los vértices de V.

Definición 26. Sea $G = (V, A)$ un grafo no dirigido, se dice que G es un **grafo bipartido** si $V = V_1 \coprod V_2$, donde V_1 y V_2 son dos conjuntos no vacíos y disjuntos y el grafo G consiste en aristas que conectan vértices de V_1 con vértices de V_2.

Definición 27. Sea $G = (V, A)$ un grafo no dirigido, se dice que es el **grafo bipartido completo** $K_{n,m}$ de $n + m$ vértices, si $V = V_1 \coprod V_2$ con $Card(V_1) = n$ y $Card(V_2) = m$, es un grafo bipartido y además todo vértice de V_1 está conectado con todos los vértices de V_2.

2.2.2. Isomorfismo de grafos.

Definición 28. Sean $G = (V, A)$ y $G' = (V', A')$ dos grafos, se dice que son **isomorfos** si existe una función biyectiva $f : V \to V'$ tal que $\{v_1, v_2\} \in A$ si y sólo si $\{f(v_1), f(v_2)\} \in A'$. De la función f se dice que es un **isomorfismo de grafos**.

Una consecuencia inmediata de la definición de isomorfismo de grafos es que dos grafos isomorfos tienen el mismo número de vértices y de aristas y que además los grados de los vértices coinciden en ambos.

Definición 29. Sea $G = (V, A)$ un grafo no dirigido es **conexo** si $\forall\, v, v' \in V$ existe un camino entre v y v'.

Cuando el grafo G no es conexo se puede representar como unión disjunta de grafos conexos $G = \coprod_i G_i$. A los grafos G_i se les llama **componentes conexas** de G.

Definición 30. Sea $G = (V, A)$ un grafo simple, se dice que un subgrafo $G' = (V', A')$ de G; (es decir, $V \subset V'$ y $A \subset A'$) es una **componente conexa** de G si es conexo y contiene todas las aristas de G que inciden con vértices de G'; es decir $\{v, u\} \in A$ y $v \in V'$ entonces $u \in V'$ y $\{v, u\} \in A'$.

Nota 4. Dos grafos isomorfos tienen el mismo número de componentes conexas.

Teorema 3. *Sea $G = (V, A)$ un grafo conexo no dirigido, entonces para cualquier par de vértices de G existe un camino simple que los une.*

Demostración. Sean $v, u \in V$ y $v = v_1, \ldots, v_n = u$ un camino entre v y u de la menor longitud posible. Por reducción al absurdo, si existen $i < j \in \{1, \ldots n\}$ tales que $v_i = v_j$, entonces $v, v_1, \ldots, v_{i-1}, v_j, v_{j+1}, \ldots, v_n$ es un camino de menor longitud del de partida y que une v y u, absurdo. □

Definición 31. Sea $G = (V, A)$ un grafo y sea $v \in V$ un vértice en G. Se dice que v es un **vértice de corte** si el grafo $G' = (V - \{v\}, A - \{\text{todas las aristas incidentes con } V\})$ tiene más componentes conexas que G.

Definición 32. Sea $G = (V, A)$ un grafo y sean $\{v_1, v_2\} \in A$ una arista de G. Se dice que $\{v_1, v_2\}$ es una **arista de corte** si el grafo $G' = (V, A - \{v_1, v_2\})$ tiene más componentes conexas que G.

Nota 5. Dos grafos isomorfos tienen el mismo número de vértices y aristas de corte así como los mismo subgrafos.

2.3. Grafos eulerianos y hamiltonianos.

Definición 33. Se dice que un camino en G es un **recorrido euleriano** si es un recorrido que pasa por todas las aristas del grafo.

Definición 34. Se dice que un camino en G es un **circuito euleriano** si es un circuito que pasa por todas las aristas de G.

Definición 35 (Grafos eulerianos)**.** Se dice que un grafo no dirigido es **euleriano** si contiene un circuito euleriano.

Teorema 4. *Sea $G = (V, A)$ un grafo no dirigido, G es un grafo euleriano si y sólo si las aristas están en la misma componente conexa y el grado de todos los vértices es par.*

Demostración. La condición suficiente es clara, pues si existe un circuito euleriano todas las aristas están conectadas y por tanto en la misma componente conexa. Además, los vértices tiene grado par puesto que el circuito al pasar por un vértice lo hace con una arista para entrar y otra para salir.

Para ver el recíproco, se construye el siguiente algoritmo.

Se puede suponer sin pérdida de generalidad que el grafo G es conexo y sin lazos ya que por hipótesis todas las aristas están en la misma componente conexa. En cuanto a los lazos, no ofrecen dificultad a la hora encontrar un circuito euleriano.

Algoritmo de borrado.

- **Entrada** $G = (V, A)$, $C = (V_C, A_C)$ C circuito de G
- $A_H = A - A_C$ y $V_{aislados}$ son los vértices aislados de (V, A_H)
- $V_H = V - V_{aislados}$
- **Salida** (V_H, A_H)

Nota 6. El grafo (V_H, A_H) es del mismo tipo que G, grafo no dirigido conexo y sin lazos y con el grado de todos los vértices un número par.

Algoritmo de insertado (C, C')**.**

- **Entrada** $G = (V, A)$ grafo, C y C' circuito de G tales que C' empieza en un vértice interior de C.

  $C := v_1, \ldots, v_n, v_1$ y $C' := u_1 = v_i, \ldots, u_m, u_1$

  $V := v_1, \ldots, v_i, u_2, \ldots, u_m, v_i, v_{i+1}, \ldots, v_n, v_1$

- $A' := \emptyset$

  - para $i = 1$ hasta $n + m + 1$

- $A' := A' \cup \{\{v_i, v_{i+1 \text{ mód } m+n}\}\}$, mód $m+n$ significa que los subíndices no mayores ni de m ni de n; es decir, los vértices son los extremos de las aristas de V, renombrando los vértices de $V := v_1 \dots v_{n+m} v_1$.

- **Salida** $(\{V\}, A')$, Por $\{V\}$ se quiere expresar el conjunto que forman los vértices por donde pasa el circuito.

Algoritmo de búsqueda de un circuito euleriano.

- **Entrada** $G = (V, A)$ grafo sin lazos y conexo
- $C :=$ circuito de G. (*)
- $H = (V_H, A_H) :=$ Borrado(C, G)
- mientras $V_H \neq \emptyset$
  - C' cualquier circuito de H con origen en un vértice interior de C
  - $H :=$ Borrado(C', H)
  - $C :=$ Insertado(C', C)
- **Salida** C

(*) Un circuito C partiendo de cualquier vértice $v \in G$ existe siempre porque el grafo es conexo y todos sus vértices tienen grado par, ya que al tomar un recorrido de longitud máxima desde v si no es circuito, entonces existe algún vértice no contenido en el camino (grado par) que contradice la maximalidad del camino. □

Proposición 3. *Sea $G = (V, A)$ un grafo no dirigido con todas las aristas contenidas en la misma componente conexa, G posee un recorrido euleriano si y sólo si todos los vértices tienen grado par excepto exactamente dos con grado impar.*

Demostración. Basta observar que $G = (V, A)$ admite un recorrido euleriano, $v, v_1, \dots, v_n, v'$, si y sólo si $G' = (V \cup \{u\}, A \cup \{u, v\} \cup \{v', u\})$, donde u es un vértice nuevo no contenido en G, admite un circuito euleriano. □

Otra fórmula para encontrar un circuito euleriano en un grafo euleriano es mediante el algoritmo de Fleury que se basa en recorrer las aristas arbitrariamente siguiendo las propiedades siguientes:

i. Se borran las aristas a medida que son atravesadas

ii. Solo se recorre una arista de corte si no queda otra alternativa

Algoritmo de Fleury

1. **Entrada** $G = (V, A)$ Grafo euleriano o con recorrido euleriano.
2. Se empieza en $u \in V$ tal que $gr(u) = 2k + 1$. Si no hay, en cualquier vértice. $C := u$
3. Si $gr(u) = 0$ parar.
4. Si $gr(u) = 1$ con $a = \{u, v\}$,
   - $G = (V, A) := (V - u, A - \{a\})$, hacer $C := Cav$. Ir al paso siguiente.

   Si $gr(u) > 1$ elegir una arista $a = \{u, v\}$, cuya eliminación no desconecte el multigrafo.
   - $G = (V, A) := (V, A - \{a\})$ y hacer $C = Cav$. Ir al paso siguiente.
5. Reemplazar u por v y volver al paso tres.

Definición 36. Se dice que un camino en G es un **ciclo hamiltoniano** si es un ciclo que pasa por todos los vértices del grafo.

Definición 37 (Grafos hamiltonianos)**.** Sea $G = (V, A)$ un grafo no dirigido, se dice que G es un **grafo hamiltoniano** si contiene un ciclo hamiltoniano.

No existen caracterizaciones de los grafos hamiltonianos. Es claro que en todo grafo hamiltoniano cada vértice tiene grado mayor o igual a dos, pero no es una condición suficiente. A pesar de todo se tiene los siguientes resultados.

Teorema 5. *Sea un grafo $G = (V, A)$ con el número de vértices $= n \geq 3$. Si el grafo verifica alguna de las condiciones siguiente, entonces es Hamiltoniano.*

i. **[Teorema de Ore]** *La suma de los grados de cualquier par de vértices no adyacentes es mayor o igual que n.*

ii. **[Dirac]** *El $gr(u) \geq n/2$ para todo $u \in V$.*

iii. El número de aristas $|A| \geq \frac{(n-1)(n-2)}{2} + 2$.

Por otra parte, Proposiciones que permiten asegurar que ciertos grafos no son hamiltonianos:

Proposición 4. *Sea $G = (VA)$ un grafo simple y conexo y $|V| \geq 3$. Si es hamiltoniano, entonces para cada subconjunto $U \subset V$ el grado obtenido al eliminar de V los vértices de U y las aristas incidentes con dichos vértices tiene a lo sumo $|U|$ componente conexas.*

Demostración. Si el grafo G es hamiltoniano, entonces contiene un ciclo hamiltoniano

$$a, v_1, v_2, \cdots, v_{n-1}, a.$$

Sea H el grafo formado por los vértices y aristas del ciclo anterior, y sea $U \subset V$. (Obsérvese que los vértices de H son los mismos que los del grafo original G). Sea k el número de componentes conexas del grafo $(V - U, E')$ donde E' es el subconjunto de aristas de G caracterizado por el hecho de que sus extremos pertenecen a $V - U$, y sea k' el número de componentes conexas del subgrafo de H obtenido al eliminar de H los vértices pertenecientes a U junto con las aristas incidentes con ellos. Evidentemente, $k \leq k'$. Puesto que el grafo H se puede representar como un ciclo, es obvio que al eliminar un punto (y las aristas incidentes con él) el nuevo grafo así obtenido es conexo (es decir, tiene una única componente conexa), al eliminar dos puntos (y las aristas incidentes con ellos) el nuevo grafo así obtenido tiene como mucho dos componentes conexas y así sucesivamente. En general si quitamos p vértices (junto con sus aristas incidentes) obtenemos un grafo con, a lo sumo, p componentes conexas. Por consiguiente $k \leq k' \leq p = |U|$. □

2.4. Grafos etiquetados.

Definición 38. Sea $G = (V, A)$ un grafo simple, se dice que es un **grafo etiquetado** si existe una función $d : A \to \mathbb{R}$ que asigna a cada arista una **etiqueta**. Este tipo de grafos se denotan por $G = (V, A, d)$

Con esta clase de grafos se puede resolver el problema de la búsqueda del camino más corto entre dos vértices.

Algoritmo de Dijkstra.

- **Entrada**: $G = (V, A, d)$ $v, u \in V$
- $L(u) := 0$; $L(x) := \infty$ para todo $x \neq u$; $T := \emptyset$

- mientras $v \in V - T$.
  - $x := a$ donde $a \in V - T$ y $L(a)$ es mínimo; $T := T \cup \{x\}$
  - para $y \in V - T$
    - si $L(x) + d(\{x, y\}) < L(y)$, entonces
      - $L(y) := L(x) + d(\{x, y\})$
      - $f(y) := x$
- **Salida:** $L(v)$ es la longitud del camino mínimo y $(v, f(v), f(f(v)), \ldots, u)$ es el camino mínimo de v a u.

2.5. Árboles.

Definición 39. Sea $G = (V, A)$ un grafo simple y no dirigido, se dice que es un **árbol** si es conexo y sin ciclos.

Proposición 5. *Sea $G = (V, A)$ un grafo simple y no dirigido. G es un árbol si y sólo si para todo $u, v \in V$ existe un único camino simple que une u y v.*

Demostración. La implicación directa es fácil. Por ser conexo existe un camino simple entre dos cualesquiera de los vértices de V, ver proposición (3). Veamos que es único. Sean $u, v \in V$ dos vértices del árbol y supongamos que existen $C = u = u_1, \ldots, u_n = v$ y $C' = u = v_0, \ldots, v_m = v$ dos caminos distintos en G y sea $n < m$. Sea $i \in \{1, \ldots, n\}$ el mínimo subíndice tal que $u_i \neq v_i$ y además existen $j_1 > i$ y $j_2 > i$ (ambos caminos terminan en v por tanto como máximo $j_1 = j_2 > i$ tales que $u_{j_1} = v_{j_2}$. Por tanto un ciclo es

$$C'' = u_{i-1}, u_i, u_{i+1}, \ldots, u_{j_1} = v_{j_2}, v_{j_2-1}, \ldots, v_{i-1} = u_{i-1}$$

Recíprocamente, la conexión se sigue de que entre cualquier par de vértices de G existe un camino simple que los une. Y la no existencia de un ciclo C contenido en G se sigue de que en caso contrario tomando un par vértices cualesquiera de C, resultan (utilizando dicho ciclo) inmediatamente dos caminos simple que unen esos dos vértices. □

Definición 40. Sea $G = (V, A)$ un árbol, se dice que es un **árbol con raíz** si existe un vértice r distinguido al que se le llama raíz. Se suele dibujar el grafo con la raíz en la parte superior y esta elección determina una dirección (alejarse de la raíz) en el grafo. Así, se dice que se *alcanza* u desde v si partiendo de uno se llega al otro vértice en la dirección que infiere la raíz.

Definición 41. Sea $T = (G, r)$ un árbol con raíz.

i. Se dice que $u \in T - \{r\}$ es **padre** de $v \in V$ si existe una arista entre u y v y se alcanza v desde u. Entonces, v es un **hijo** de u.

ii. Los **antecesores** de un vértice v son el resto de los vértices que se encuentran en el camino que une v con la raíz r. Es decir, todo vértice alcanzable desde v es un descendiente suyo. Así mismo, se dice que v es un **descendientes** de todos sus antecesores.

iii. Se denominan **hojas** o vértices colgantes o terminales a los vértices de G distintos de r que tienen grado 1.

iv. Se denomina **profundidad** de $u \in G$ a la longitud del camino que une u con la raíz r. Y **altura** a la mayor profundidad del vértice raíz r.

v. Se llaman vértices internos a todos los que no son hojas.

vi. La raíz está a nivel cero y las hojas que dan las altura a nivel máximo.

Definición 42. Sea $T = (G, r)$ un árbol con raíz.

i. Se denomina **subárbol** con raíz v si se toma el vértice v y todos sus descendiente así como las aristas que unen este con sus descendientes.

ii. Se dice que $T = (G, r)$ es un árbol con raíz m**-ario** si todos sus vértices tienen a lo sumo m hijos. Si ocurre que todos los vértices salvo las hojas tienen exactamente m hijos se dice que el árbol con raíz es m**-ario completo**. Si $m = 2$ se denominan **árboles binarios**

Proposición 6. *Un árbol con n vértices tiene $n - 1$ aristas.*

Demostración. Sea $T = (V, A)$ un árbol con n vértices y tómese un vértice r de T como raíz. Se define la función $A \to V - \{r\}$ como aquella que asigna a cada arista el vértice final considerando el sentido que define la elección de un vértice como raíz. La función es biyectiva y eso demuestra que $|V| - 1 = |A|$. □

Proposición 7. *Cualquier grafo conexo (respect. sin ciclos) con n vértices y $n - 1$ aristas es un árbol.*

2.5.1. Árboles de búsqueda binarios.

Algoritmo de búsqueda binaria. Resuelve problemas del tipo:
"Ordenar totalmente una serie de datos para una fácil localización"

Proposición 8. *Sea (T, r) un árbol binario. Este define una relación binaria de orden total de la siguiente manera: " $v \in V$ es mayor que todos sus descendientes de la izquierda y menor que sus descendientes de la derecha."*

Recíprocamente,

Proposición 9. *Sea X un conjunto y R una relación de orden total en X. Entonces se puede construir un árbol binario iterativamente. Se coloca el primer vértice que se convertirá en la raíz r del árbol. El siguiente punto de X será el vértice hijo de r de la izquierda si es menor que r y de la derecha si es mayor. Los siguientes puntos de X se colocan en el árbol comenzando de nuevo por la raíz y moviéndose a la derecha si es mayor y a la izquierda si es menor. Si en algún momento no existiese un hijo con quien comparar, el nuevo punto se convertiría en ese hijo. Finalmente se puede elegir otro elemento raíz si el árbol no tiene las ramas con una longitud similar.*

2.5.2. Árboles de decisión.

Otra aplicación de los árboles con raíz es la solución de problemas del tipo modelizar la toma de decisiones. Cada vértice interno corresponde a una decisión y el subárbol colgado de él corresponde a las soluciones alternativas posibles. Las diferentes maneras de resolver el problema se corresponden con los caminos que permiten recorrer el árbol desde la raíz a las hojas.

2.5.3. Árboles generadores.

Sea $G = (V, A)$ un grafo simple.

Definición 43. Un **subgrafo generador** es un subgrafo de G que contiene todos los vértices de G.

Definición 44. Sea $G' = (V, A')$ un subgrafo generador de G. Se dice que G' es un **subárbol generador** si es un árbol.

Proposición 10. *Sea $G = (V, A)$ un grafo simple. Existe un subárbol generador de G si y sólo si G es conexo.*

Demostración. La implicación directa es clara por un árbol generador es un grafo conexo.

La forma de construir un árbol generador a partir de un grafo conexo es iterativamente localizar un ciclo en G y eliminar una arista. Siguiendo este proceso se eliminan uno a uno los ciclos de G y se preserva la conexión. □

La localización de ciclos en grafos conexos que los tienen no siempre es fácil, de modo que otro método de demostrar la proposición anterior, es fijar un vértice arbitrario v e ir añadiéndole aristas y vértices convenientemente y "conexamente". Se describen a continuación dos formas de realizar esto.

búsqueda en amplitud. Se trata de recorrer todos los vértices adyacentes a v y las aristas correspondientes. Se repite el proceso con los vértices acabados de añadir y así sucesivamente hasta que se recorran todos los vértices de G.

Algoritmo de búsqueda de amplitud

- **Entrada**: $G = (\{u = v_0, v_1, \ldots, v_n\}, A)$ grafo conexo, $u \in V$ vértice inicial. $V' := \emptyset$, $A' := \emptyset$ $C : 0 = \{u\}$.
- $V' := \{u\}$, $V := V - \{u\}$. Mientras V sea distinto del vacío.
  - $i = 0$, S :=conjunto de vértices adyacente a v_i para algún vértice de V'. Si no es vacío, $j = 1, \ldots j(i)$, y los elementos de S se numeran con u^i_j. Entonces $j := j + 1$, $V' := V' \cap \{u^i_j\}$, $A' := A' \cap \{v_i, u^i_j\}$ y $C := Cu_j$, $V := V - \{u_j\}$.
  - Si $S = \emptyset$ se termina.
  - $i := i + 1$
- **Salida** C camino que recorre todos los vértices de G y $G' = (V', A')$ árbol generador de G.

búsqueda en profundidad. Ahora se añade a v una sola arista incidente y el extremo a ésta. Se hace lo mismo con el último extremo añadido y se continúa así hasta terminar con los vértices.

Algoritmo de búsqueda de profundidad

- **Entrada**: $G = (V, A)$ grafo conexo, $u \in V$ vértice inicial.
- $V' := \{u\}$, $V := V - \{u\}$, $A' := \emptyset$
  - mientras V no esté vacío, no parar.
  - Si u tiene vértices adyacentes, se añade uno, $u_1 \in V - V'$ a V' y la arista que los une a A' y $C := uu_1$ y se considera u_1 como nuevo vértice en busca de adyacentes.
  - Si no hay vértices adyacente, se vuelve al padre del último vértice añadido y se repite el proceso.
- **Salida** C es el camino por búsqueda de profundidad y $G' = (V', A')$ es un árbol generador de G.

Definición 45. Sea $G = (V, A, d)$ un grafo etiquetado y conexo. Un árbol generador en G es un **árbol generador mínimo** si la suma de sus aristas es la más pequeña posible.

Existen dos algoritmos principalmente para calcular árboles generadores mínimos.

Algoritmo de Prim.

- **Entrada**: $G = (V, A)$; $u \in V$; $V(T) := \{u\}$; $A(T) := \emptyset$
- mientras $V(T) \neq V$
  - Añadir a $A(T)$ una de las aristas de menor peso incidente con un vértice de $V(T)$ (y sólo uno) y añadir su extremo a $V(T)$ (sin formar ciclos).
- **Salida**: $(V(T), A(T))$

Algoritmo de Kruskal.

- **Entrada**: $G = (V, A)$; $u \in V$; $V(T) := \{u\}$; $A(T) := \emptyset$
- mientras $V(T) \neq V$ y $|A(T)| \neq |V| - 1$
  - Añadir a $A(T)$ las aristas (una a una) de menor peso siempre que no forme un ciclo con las aristas de $A(T)$ y añadir sus extremos a $V(T)$
- **Salida**: $(V(T), A(T))$

2.6. Coloración de grafos.

Definición 46. Sea $G = (V, A)$ un grafo, $\lambda \in \mathbb{N}$ y $X_\lambda = \{1, \ldots, \lambda\}$. Una **coloración con λ colores** del grafo G es una aplicación $f : V \to X_\lambda$ tal que si $u, v \in V$ y $\{u, v\} \in A$, entonces $f(u) \neq f(v)$.

Definición 47. Sea $G = (V, A)$ y $\lambda \in \mathbb{N}$, se denota por $P_G(\lambda) = [G]_\lambda$ al número de coloraciones que admite G con λ colores.

Proposición 11. *Sea $G = (V, A)$ un grafo, si λ varía en $\mathbb{N}$, $P_G(\lambda)$ es un polinomio denominado* **polinomio cromático** *y como dice la definición anterior, para cada λ da el número de coloraciones con λ colores.*

Ejemplo 1.

i. Si G tiene n vértices aislados, $P_G(\lambda) = \lambda^n$

ii. Si $G = P_n$, entonces $P_G(\lambda) = \lambda(\lambda - 1)^{n-1}$

iii. Si $G = K_n$, entonces $P_G(\lambda) = \lambda(\lambda - 1) \cdots (\lambda - (n - 1))$

iv. Si G es un árbol, entonces $P_G(\lambda) = \lambda(\lambda - 1)^{n-1}$

v. Si $G = G_1 \sqcup G_2$, entonces $P_G(\lambda) = P_{G_1}(\lambda) \cdot P_{G_2}(\lambda)$.

Lema 4. *Sea $a = \{u, v\} \in A$ una arista de un grafo $G = (V, A)$, entonces $P_G(\lambda) = P_{G-a}(\lambda) - P_{G/a}(\lambda)$, donde $G - a$ es el grafo que resulta de eliminar la arista a del grafo G y G/a el grafo resultante de identificar u con v y de olvidar la arista a; es decir, contraer la arista a a un punto.*

Demostración. Basta observar que

$$\begin{aligned} \{\text{coloraciones de } G - a\} &= \{\text{coloraciones de } G \text{ tales que } u \text{ tiene el mismo color que } v\} + \\ &+ \{\text{coloraciones que colorean } u \text{ con distinto color que } v\} = \\ &= P_{G/a}(\lambda) + P_G(\lambda). \end{aligned}$$

□

Demostración proposición (11). Por inducción sobre el número m de aristas.

Si $m = 0$ y el grafo G tiene n vértices entonces $P_G(\lambda) = \lambda^n$ es un polinomio.

Si $m > 0$ y se supone que el resultado es cierto para los grafos con $m - 1$ aristas o menos, entonces sea $a = \{u, v\} \in A$ una arista de G. Por el lema anterior, $P_G(\lambda) = P_{G-a}(\lambda) - P_{G/a}(\lambda)$ y además, $G - a$ y G/a son grafos con menos de m aristas. Luego, por hipótesis de inducción, son polinomios. Como la suma de polinomios es un polinomio, se termina. □

Lema 5. *Sea $u, v \in V$ dos vértices de un grafo $G = (V, A)$ tales que $\{u, v\}$ no es una arista de G. Entonces $P_G(\lambda) = P_{G+a}(\lambda) + P_{G \cup a}(\lambda)$, donde $G + a$ es el grafo que resulta de añadir la arista a del grafo G y $G \cup a$ el grafo resultante de identificar u con v y de olvidar la arista a; es decir, contraer la arista a a un punto.*

Definición 48. Sea $G = (V, A)$ un grafo, se dice que G es **λ-coloreable** si admite una coloración con λ colores.

Definición 49. Se llama *número cromático* al menor número $\lambda \in \mathbb{N}$ tal que G es λ-coloreable y se denota $\chi(G)$.

Proposición 12. *El número cromático es el menor número natural para el cual el polinomio cromático tiene valor positivo.*

Proposición 13. *Sea $G = (V, A)$ un grafo tal que*

$$G = G_1 \cup G_2 \qquad y \qquad G_1 \cap G_2 = K_n$$

para algún $n \in \mathbb{N}$, entonces

$$P_G(\lambda) = \frac{P_{G_1}(\lambda) \cdot P_{G_2}(\lambda)}{P_{K_n}(\lambda)}.$$

Demostración. Se demuestra por inducción sobre el número de aristas más el número de vértices= $n = |V| + |A|$.

Si $n = 1$ no hay nada que probar porque en ese caso es un punto.

Supóngase que el teorema es cierto para los grafos para los cuales el número de vértices más el número de aristas es $n - 1$. Sea entonces G un grafo con $|V| + |A| = n$. Sea $G = G_1 \cup G_2$ con $G_1 = (V_1, A_1)$ y $G_2 = (V_2, A_2)$. .

Caso 1. Supóngase que $A_1 \subset A_2$. Si $V_1 \subset V_2$, entonces $G_1 \subset G_2$ y se concluye.

Si $V_1 \not\subseteq V_2$, entonces $\exists v \in V_1 - V_2$ y es aislado. Sea $G_1' = (V_1 - \{v\}, A_1')$ el grafo que resulta de eliminar de G_1 el vértice v y sea $G' = G_1' \cup G_2$. G' verifica que $G_1' \cap G_2 = K_n$ pues como $v \notin V_2$ ninguna arista eliminada pertenecía a K_n y se puede aplicar hipótesis de inducción. Así,

$$G' = \frac{[G_1'][G_2]}{[K_n]} \quad \text{y} \quad \frac{[G]}{[\cdot]} = \frac{\frac{[G_1]}{[\cdot]}[G_2]}{[K_n]}$$

Caso 2. Si $A_1 \subsetneq A_2$, entonces $\exists a = \{x, y\} \in A_1 - A_2$. Como $a \in A_1$ entonces $x, y \in V_1$ y por otro lado, si $x, y \in G_1 \cap G_2 = K_n$, como es completo también la arista a pertenece a $G_1 \cap G_2$, luego $a \in A_2$, contradicción.

Por tanto, existen $x, y \in V_1$, $x \notin V_2$, $a \in A_1$ y $a \notin A_2$. Utilizando el teorema para eliminar la arista a e identificar los vértices x e y, se concluye. En efecto, basta observar que $G_{a^-} = G_{1a^-} \cup G_2$ y $G_- = G_{1a^-} \cup G_2$, que $G_{1a^-} \cap G_2 = K_n$ y $G_{1a^-} \cap G_2 = K_n$ y que de $[G] = [G_{a^-}] - [G_-]$ se sigue que

$$[G] = \frac{[G_{1a^-}][G_2]}{[K_n]} - \frac{[G_{1a^-}][G_2]}{[K_n]} = \frac{[G_1][G_2]}{[K_n]}.$$

□

2.7. Grafos Planos.

Definición 50. Sea $G = (V, A)$ un grafo, se dice que es un grafo **plano** si admite una representación gráfica en el plano de modo que cada arista corta únicamente a otra arista en un vértice que sea extremo de ambas.

Definición 51.

1. Sea $G = (V, A)$ un grafo plano. Una representación de este en el plano con la condición que sus aristas se corten solamente en sus vértices se denomina **mapa**.
2. Se dice que el **mapa** es **conexo** si el grafo que determina es conexo.
3. Se denomina **región o cara** del grafo a cada una de las partes conexas del plano que determinan las aristas y los vértices de G.

Definición 52. Sea $G = (V, A)$ un grafo, se denomina **grado de una región** del grafo a la longitud del camino que la bordea.

Teorema 6. *La suma de los grados de las regiones de un mapa es igual al doble del número de aristas del grafo simple que representa.*

Demostración. Cada arista aparece en el borde de exactamente dos regiones. □

Teorema 7 (Teorema de Euler)**.** *Sea M un mapa conexo, sea $|R|$ el número de regiones que representa el grafo simple $G = (V, A)$, entonces $|R| - |A| + |V| = 2$.*

Demostración. Se demuestra por inducción sobre el número de aristas, $|A|$, de G.

Si $|A|$ es cero, entonces como M es conexo, tiene que ser $(\{v\}, \emptyset)$ y en el mapa hay solo una región. Por lo tanto $1 - 0 + 1 = 2$.

Supongamos ahora que $|A| \geq 1$ y que la fórmula es cierta para mapas con un número menor de aristas.

Caso 1. G tiene un ciclo. Considérese $G' = (V, A - \{e\}) \subset G$ donde $\{e\}$ es una arista de un ciclo de G. Considérese además el mapa que representa este subgrafo. Así por hipótesis de inducción sobre G' (G' sigue siendo plano y conexo puesto que la arista eliminada pertenece a un ciclo y no es arista de corte) y tiene exactamente una arista y una región menos que G. Así la fórmula es cierta para M' mapa de G' y se sigue que $|R| - 1 - (|A| - 1) + |V| = 2$ y por tanto

$$|R| - |A| + |V| = 2.$$

Caso 2. G es un árbol. Entonces existe al menos un vértice hoja v. Sea $\{w, v\}$ la única arista incidente con v de G y considérese $G' = (V - \{v\}, A - \{w, v\})$ y aplíquese de nuevo la hipótesis de inducción. Se M' un mapa de G' y obsérvese que el número de regiones de M' coincide con el número de regiones de M. Entonces $|R| - (|A| - 1) + |V| - 1 = 2$ de donde se sigue el resultado para G.

□

Corolario 2. *Sea $G = (V, A)$ un grafo simple conexo plano con $|V| > 2$. Entonces*

$$|A| \leq 3|V| - 6.$$

Demostración. Sea M un mapa de G con $|R|$ regiones. Como el grafo es simple, el grado de cualquier región es al menos tres, luego

$$2|A| \geq 3|R|,$$

y por tanto

$$2 = |V| - |A| + |R| \leq |V| - \frac{1}{3}|A|.$$

De lo que se deduce que $|A| \leq 3|V| - 6$. □

Corolario 3. *Sea $G = (V, A)$ un grafo simple plano y conexo con $|V| > 2$. Si G no posee ningún subgrafo isomorfo a K_3, entonces*

$$|A| \leq 2|V| - 4$$

.

Demostración. Como G no contiene un K_3, el grado de cada región es al menos 4, entonces como se hizo en el corolario anterior $2 \leq |V| - \frac{1}{2}|A|$ y se termina. □

Proposición 14. *Los grafos K_5 y $K_{3,3}$ no son planos.*

Demostración. El grafo completo K_5 tiene 5 vértices y 10 aristas, y como $|A| = 10 > 9 = 3|V| - 6$ se contradice la anterior proposición.

El grafo bipartido $K_{3,3}$ tiene 6 vértices y 9 aristas con grado 3, por tanto $3|R| = 2 \cdot 9 = 18$. Por otro lado, el teorema de Euler dice que $3|R| = 3(2 - |V| + |A|) = 3(2 - 6 + 9) = 15$. Â¡Contradicción! □

Definición 53. Sea $G = (V, A)$ un grafo, sean $u, v \in V$ vértices de V y $\{u, v\} \in A$ una arista de G. Se dice que el nuevo grafo $G' = (V \cup \{w\}, (A - \{u, v\}) \cup \{u, w\} \cup \{w, v\})$ es una **subdivisión elemental** de G.

Teorema 8 (Teorema de Kuratowski.)**.** *Un grafo simple G es plano si y sólo si no contiene ningún subgrafo que sea isomorfo a una subdivisión elemental de K_5 o a $K_{3,3}$.*

Definición 54. Sea $G = (V, A)$ un grafo, se dice que dos regiones de G son **adyacentes** si las separa una arista.

Teorema 9. *Todo grafo plano admite una coloración con cuatro colores.*

Demostración. Sea G un grafo simple y plano y sea M un mapa que representa a este. Sea G' el grafo dual a G (el grafo dual consiste en transformar las regiones del grafo original en vértices del dual de modo que si dos regiones son adyacentes se conviertan en vértices adyacentes). Colorear G es lo mismo que pintar las regiones de G'. Tras más de 100 años de investigación, en 1977 "K. Appel, W. Haken y J. Koch" demostraron una solución al siguiente **Teorema de los Cuatro Colores**: "Se puede colorear cualquier mapa (usual) con cuatro colores diferentes de modo que no haya dos regiones adyacentes con el mismo color". □

Teorema 10. *Un grafo es bipartido si y sólo si se puede colorear con dos colores.*

Demostración. Un grafo $G = (V, A)$ es bipartido si el conjunto de vértices $V = V_1 \coprod V_2$ se puede escribir como unión disjunta de vértices de forma que las arista de G no unen vértices de un mismo conjunto. Es claro ahora que si $f : V \to \{0, 1\}$ es una coloración en G, se tiene que "$v \in V_1$ si y sólo si $f(v) = 0$ y $v \in V_2$ si y sólo si $f(v) = 1$" determina tanto la coloración como los subconjuntos V_1 y V_2. □

Corolario 4. *Un grafo es bipartido si y sólo si no tiene ciclos con longitud impar.*

Demostración. La implicación directa es clara puesto que si $f : V \to \{0, 1\}$ es una coloración de G, los vértices consecutivos de cualquier ciclo en G tienen que tener alternativamente los colores 0 y 1, salvo el primero y el último que coinciden. Por tanto, el número de aristas ha de ser par.

Recíprocamente, si G es un grafo en el que todos los ciclos tienen longitud par. Encontremos una coloración por inducción sobre el número de aristas $|A|$.

Si $|A| = 0$ no hay nada que probar.

Supongamos que el teorema es cierto para los grafos con menos de n aristas y sea $G = (V, A)$ con $|A| = n$. Sean $u, v \in V$, $\{u, v\} \in A$ y considérese $G' = (V, A - \{u, v\})$.

Caso 1. Si u y v están en componentes conexas distintas de G', sean G'_u y G'_v están respectivamente. Como G' tiene $n-1$ aristas se puede aplicar hipótesis de inducción y encontrar una coloración con dos colores $f : G' \to \{0, 1\}$. Si ésta colorea u y v con colores distintos, es también una coloración de G y se termina. Si $f(u) = f(v)$, sea $g : G \to \{0, 1\}$ definida por

$$g(x) = (f(x) + 1) \text{ mód } 2 \text{ si } x \text{ es un vértice de } G_u$$

$$g(x) = f(x) \text{ si } x \text{ es un vértice de } G' \text{ que no está en } G_u$$

es una coloración de G por que sólo se han cambiado todos los vértices de una componente conexa de G.

Caso 2. Si u y v están en la misma componente conexa de G', existe un camino simple $uv_1, \cdots v_n v$ entre u y v contenido en la componente conexa. Como además uv está en esta misma componente conexa, $uv_1, \cdots v_n vu$ es un ciclo en G y por tanto tiene longitud par. Y $uv_1, \cdots v_n v$ tiene longitud impar.

Por hipótesis de inducción el grafo G' admite una coloración $f : V \to \{0, 1\}$. Como además el camino $uv_1, \cdots v_n v$ tiene longitud impar, $f(u) \neq f(v)$ y por tanto f es también una coloración de G.

□

Capítulo 3

Teoría de Códigos.

3.1. Introducción.

Los lenguajes naturales son códigos pero no corrigen errores y los detectan mal. Casi siempre, por sintaxis o contexto, podemos detectar un error, pero es muy difícil corregirlo.

Ejemplo 2.

- Una cita;
- en la habitación una nota

"Xo te amo";

- Se detecta el error, pero duda entre:

"Yo te amo"ó "No te amo".

- En inglés, "I love Xou", con las posibles interpretaciones "I love You"ó "I love Lou"(quien además podría ser su amigo. . .)

Â¿Porqué los códigos? Para poder resolver estas dudas, apelaremos a los códigos y trataremos también de mostrar para qué sirven y cuál es el interés en ellos.

El objetivo es dar una breve introducción a la teoría de los códigos, a través de numerosos ejemplos y de una clase particular muy rica de códigos, los códigos lineales.

Proceso de comunicación. En el proceso de comunicación (en cualquier sentido) se producen los siguientes pasos:

- Envío de un mensaje (éste se hace por medio de un canal de comunicación; (con posibilidad de ruidos, fallos de comunicación, falta de datos, de imagen, etc $\cdots$)).
- Traducción, entre mensaje original (*o palabra fuente*) x y el tipo de mensaje c que el canal está habilitado para enviar (*palabras código*). **codificación.**
- Envío y recepción del mensaje codificado (*palabra recibida*) c'.
- Detección de errores (posiblemente a causa del ruido, interferencias,...) y corrección al mensaje original x' si es posible. A este proceso se le llama **decodificación**.

Esquema.

$$\begin{array}{ccc} \text{Emisor} & & \\ \downarrow x & & \\ \text{Codificación} & & \\ \downarrow c & & \\ \text{Canal} & \leftarrow & \text{Ruido} \\ \downarrow c' & & \\ \text{Decodificación} & & \\ \downarrow x' & & \\ \text{Receptor} & & \end{array}$$

En general, $x' \neq x$ y se querría que el código detectase este error y posiblemente corregirlo.

Ejemplo 3. Se trata de un vehículo de exploración en la superficie de Marte, que llamaremos $Mar\theta$, se maneja por control remoto transmitiendo impulsos eléctricos de dos voltajes distintos, que denotamos simplemente por 0 y 1, respectivamente. El vehículo se mueve un metro cada vez, en una de las cuatro direcciones posibles: norte (N), sur (S), este (E) y oeste (O). Luego, nuestros mensajes son N, S, E y O y los codificamos, por ejemplo, como 00, 11, 01 y 10. Es decir,

$$N \to 00; O \to 10; S \to 11; E \to 01$$

Ahora, supongamos que nuestro vehículo se encuentra orientado hacia el norte, al borde de un enorme cráter, con el precipicio a su derecha (o sea, hacia el este). Gráficamente,

$$\begin{array}{ccc} & \surd & \\ \surd & Mar\theta & x \\ & \surd & \end{array}$$

Enviamos el mensaje 00, es decir "avance un metro hacia el norte".

Una interferencia en la transmisión hace que $Mar\theta$ reciba 01 y se precipite en el cráter.

El problema está en nuestro código

$$C_1 = \{00, 01, 10, 11\}$$

que **no detecta errores**.

Es decir, si hay un error en la transmisión, la palabra recibida es otra palabra código.

Más precisamente, si cometemos un error al enviar 00 recibimos 01 ó 10, y como ambas son palabras de C_1, no detectamos ningún error. Análogamente para $01, 10$ y 11.

Representemos este hecho por

$$00 \to \left\{ \begin{array}{l} 10 \in C_1 \\ 01 \in C_1 \end{array} \right. , 01 \to \left\{ \begin{array}{l} 00 \in C_1 \\ 11 \in C_1 \end{array} \right. ,$$

$$10 \to \left\{ \begin{array}{l} 00 \in C_1 \\ 11 \in C_1 \end{array} \right. , 11 \to \left\{ \begin{array}{l} 01 \in C_1 \\ 10 \in C_1 \end{array} \right.$$

Esto se debe a que C_1 consta de todas las palabras de longitud 2 que se pueden formar con ceros y unos.

Ejemplo 4. Código de paridad. La forma más fácil de arreglar esto es agregar redundancia mediante un dígito de control de paridad.

Se añade un dígito extra a cada palabra código de modo que la suma de los dígitos igual a uno de cada palabra código sea par. En nuestro caso, el nuevo código es

$$C_2 = \{000, 011, 101, 110\} \subset Z_2^3.$$

Si ahora se transmite 000 y si se comete un error en la transmisión, entonces se recibe 100, 010 ó 001. Como ninguna de estas palabras pertenece al código, detectamos un error. Retransmitiendo se espera conseguir el mensaje original. Así, nuestro vehículo no se mueve.

En símbolos,

$$000 \to \begin{cases} 100 \notin C_2 \\ 010 \notin C_2 \\ 001 \notin C_2 \end{cases}, \; 011 \to \begin{cases} 011 \notin C2 \\ 111 \notin C2 \\ 010 \notin C2 \end{cases}$$

$$101 \to \begin{cases} 001 \notin C2 \\ 111 \notin C2 \\ 100 \notin C2 \end{cases}, \; 110 \to \begin{cases} 010 \notin C2 \\ 100 \notin C2 \\ 111 \notin C2 \end{cases}$$

Se dice entonces que el código C_2 detecta un error o que es 1-detector.

Nota 7. El código C_2 no detecta 2-errores. Es decir, si se cometen 2-errores (dos variaciones entre el mensaje codificado y el recibido), la palabra recibida está en el código. Sin embargo, el código C_2 no corrige ninguno de estos errores.

Esto es, una vez detectado el error, no se puede decidir cuál fue la palabra código enviada.

En nuestro ejemplo, supongamos que enviamos 000 y recibimos 010. Si bien $Mar\theta$ detecta el error, si quisiera tomar una decisión por sí mismo, éste no podría. En efecto, suponiendo un error, la palabra 010 puede ser decodificada como 110, como 000 ó como 011, todas palabras códigos.

Es posible mejorar la situación haciendo que $Mar\theta$ decida por sí mismo. Una solución fácil es agregar mayor redundancia, a costa de tener que transmitir más y perder un poco más tiempo.

Ejemplo 5. Código 3-repetición. El código siguiente

$$C_3 = \{000000, 010101, 101010, 111111\}$$

generado repitiendo tres veces cada par de dígitos del código original C_1, tiene mejores propiedades (aunque requiere más tiempo en la transmisión).

En concreto, detecta 2-errores y corrige 1-errores (veámoslo).

Ahora, si mandamos nuestro mensaje 000000 y se comete un error, cualquier palabra que nos llegue puede ser corregida. El vehículo se para y corrige el error moviéndose al lugar correcto o pidiendo que se retransmita.

Por ejemplo,

$$\begin{array}{ccc} 00 & \underset{\text{codificación}}{\longrightarrow} & 000000 \\ & & \\ \underset{\text{error peso}1}{\longrightarrow} & \begin{cases} 100000 \\ 010000 \\ 001000 \\ 000100 \\ 000010 \\ 000001 \end{cases} & \underset{\text{corrección}}{\longrightarrow} \\ & & \\ 000000 & \underset{\text{decodificación}}{\longrightarrow} & 00 \end{array}$$

De modo que si enviamos 000000 y recibimos 000100 no sólo detectamos el error sino que podemos corregirlo.

Intuitivamente, 000100 "está más cerca" de 000000 que de 010101, 101010 ó 111111. Luego, corregimos 000100 como 000000 y no como 010101 ya que es más probable cometer un error que cometer tres.

Que se haya mandado 101010 ó 111111 es mucho más improbable. Este nuevo código C_3 detecta hasta errores de peso 2 y corrige 1. Luego, hemos mejorado las propiedades de detección y de corrección del código original C_1 y de C_2.

A veces no es posible pedir retransmisión de mensajes. Se necesita que el código sea capaz de corregir errores.

Ejemplos de esto se dan en la transmisión de fotografías desde el espacio tomadas por sondas espaciales, al escuchar discos compactos o al ver DVD's, al realizar ciertas transmisiones vía satélite, etcétera.

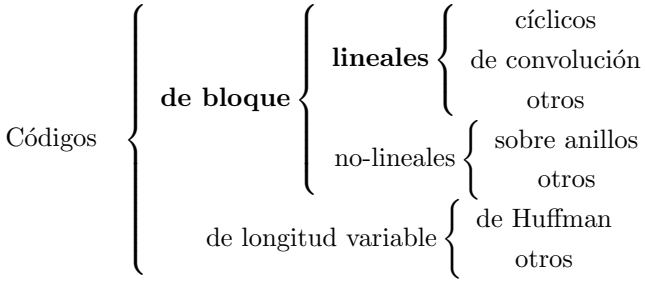
Objetivo. Construir "buenos códigos" que:

- permitan codificar gran cantidad de mensajes (*códigos de gran tamaño*).
- se transmitan con rapidez y eficacia (*alta tasa de información*).
- detecten y corrijan simultáneamente la mayor parte de los errores (*distancia mínima grande*).
- faciliten algoritmos fáciles y efectivos de (*decodificación*).

Los objetivos son contradictorios lo cual implica que se requiera un equilibrio entre ellos.

Hay muchos tipos de códigos autocorrectores.

La clasificación más básica, teniendo en cuenta la estructura del código, es la siguiente:



Las familias más conocidas de estos códigos son:

- *Códigos lineales: de Hamming*, de Hamming extendidos, simplex, de Golay, de Reed- Muller, de Goppa geométricos.
- Códigos cíclicos: BCH, de Reed-Solomon, de residuos cuadráticos, de Goppa clásicos.
- Códigos no-lineales: Hadamard, Kerdock, Justesen, Preparata.

Sin duda, los códigos cíclicos y los de convolución son los más importantes por su sencillez y utilidad. Estos códigos, no sólo poseen buenas propiedades generales y algoritmos eficientes de codificación y decodificación, sino que pueden ser implementados en computadoras a través de ciertos circuitos lineales llamados shift-registers. No es de extrañar entonces que estos sean uno de los más utilizados en la práctica.

En este curso nos interesaremos exclusivamente por los códigos lineales y sólo veremos en algún detalle los códigos de Hamming.

3.2. Códigos bloque.

Definición 55. Se definen los siguiente conceptos:

(1) Un *alfabeto* es un conjunto finito $\mathcal{A} = \{a_1, \ldots, a_q\}$.

(2) A los elementos de $\mathcal{A}$ se les llama *símbolos* y el número q es la *raíz* de $\mathcal{A}$.

(3) Una n-**cadena o palabra de longitud** n sobre $\mathcal{A}$ es una sucesión de n elementos de $\mathcal{A}$.

Así

$$a = a_{i_1} a_{i_2} \cdots a_{i_n}, \text{ con } a_{i_k} \in \mathcal{A},$$

es un n-cadena en $\mathcal{A}$ escrita por yuxtaposición.

Es habitual ver las palabras también escritas en notación vectorial

$$a = (a_{i_1} a_{i_2} \cdots a_{i_n}).$$

Definición 56. Se denota

(1) por $\mathcal{A}^n$ al conjunto de todas las n-cadenas o palabras de longitud n;

(2) por $\mathcal{A}^*$ al conjunto de todas las palabras sobre $\mathcal{A}$.

(3) Así $\mathcal{A}^* = \cup_{n\in\mathbb{N}} \mathcal{A}^n$.

Definición 57. Sea $\mathcal{A} = \{a_1, \ldots, a_q\}$ un alfabeto,

(1) un *código q-ario* sobre $\mathcal{A}$ es un subconjunto $C \subset \mathcal{A}^*$.

(2) Los elementos de C se llaman *palabras código* (codewords)

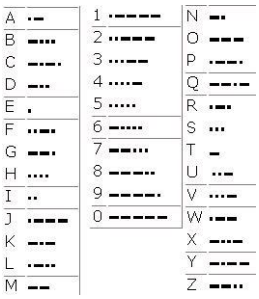
(3) El número $M = |C|$ se llama el *tamaño* del código.

(4) Si todas las palabras códigos tienen longitud fija n decimos que C es un **código de bloque** con parámetros (n, M) o que C es un (n, M)-código.

(5) Si C no es de bloque decimos que C es de **longitud variable.**

Ejemplo 6. Un ejemplo de código de longitud variable es $C = \{0, 10, 101, 1110, 11111\}$.

El ejemplo más famoso de este tipo es sin duda el código Morse.



Nota 8. No nos ocuparemos de estos códigos, de modo que por código entenderemos *código de bloque.*

Sea C un código q-ario. Se dice que C es un código binario, ternario o cuaternario según sea $q = 2$, $q = 3$ ó $q = 4$, respectivamente.

Los código binarios son los más comunes y los más viejos.

Definición 58. La **tasa de información** de un (n, M)-código q-ario se define por

$$R = R_q(C) = \frac{log_q(M)}{n}.$$

Esta tasa es la relación entre el número dígitos del mensaje original y del mensaje transmitido. Se buscan códigos con tasa de información alta, digamos $R > \frac{2}{3}$ ó $R > \frac{3}{4}$.

Ejemplo 7. Sea $\mathbb{Z}_2 = \{0, 1\}$ y sean los códigos binarios de bloque sobre $\mathbb{Z}_2$

$$\begin{aligned}(1,2) - \text{código} &= C_1 = \{0,1\} = \mathbb{Z}_2 \\ (2,3) - \text{código} &= C_2 = \{00, 01, 10\} \subset \mathbb{Z}_2^2 \\ (3,2) - \text{código} &= C_3 = \{000, 111\} \subset \mathbb{Z}_2^3 \\ (3,4) - \text{código} &= C_4 = \{000, 011, 101, 110\} \subset \mathbb{Z}_2^3\end{aligned}$$

Las correspondientes tasas de información son

$$\begin{aligned}R(C_1) &= log_2(2) = 1 \\ R(C_2) &= \frac{log_2(3)}{2} \approx 0,7925 \\ R(C_3) &= \frac{log_2(2)}{3} = \frac{1}{3} \\ R(C_4) &= \frac{log_2(4)}{3} = \frac{2}{3}\end{aligned}$$

Sea ahora $\mathbb{Z}_3 = \{0, 1, 2\}$ y sean los código ternarios de bloque

$$\begin{aligned}(3,9) - \text{código} &= C_5 = \\ &= \{001, 010, 012, 021, 100, 101, 120, 221, 222\} \\ (5,3) - \text{código} &= C_6 = \{00000, 11111, 22222\}\end{aligned}$$

con tasas de información

$$\begin{aligned}R(C_5) &= \frac{log_3(9)}{3} = \frac{2}{3} \\ R(C_6) &= \frac{log_3(3)}{5} = \frac{1}{5},\end{aligned}$$

respectivamente.

Nota 9. Si $\mathcal{A}_q \subset \mathcal{A}_r$, con $q < r$, todo código q-ario C sobre $\mathcal{A}_q$ puede pensarse como un código r-ario sobre $\mathcal{A}_r$, en cuyo caso los parámetros (n, M) de C no cambian pero su tasa de información empeora, ya que

$$R_r(C) = \frac{log_r(M)}{n} < \frac{log_q(M)}{n} = R_q(C)$$

Luego, si el alfabeto no se da explicitamente, consideramos al código C sobre $\mathcal{A}_q$ con el menor q posible, pues así tiene la máxima tasa de información.

Definición 59. Sean x e y dos palabras de igual longitud sobre el mismo alfabeto $\mathcal{A}$. La **distancia de Hamming** entre x e y, denotada por $d(x, y)$, se define como el número de coordenadas en que x e y difieren.

$$d : \mathcal{A}_n \times \mathcal{A}_n \to [0, n] \subset \mathbb{N},$$

donde $d(x, y) = \sharp\{1 \leq i \leq n : x_i \neq y_i\}$.

Muchas veces para codificar y decodificar e útil que el alfabeto $\mathcal{A}$ tenga una cierta estructura algebraica. Es habitual tomar un cuerpo finito $\mathcal{A} = \mathbb{F}_q$ como alfabeto.

El cuerpo finito $\mathbb{F}_q$ es único salvo isomorfismo y se tiene que $q = p^r$ para algún primo p y $r \in \mathbb{N}$. Si $q = p$, entonces $\mathcal{A} = \mathbb{Z}_p$, el cuerpo de enteros módulo p.

El conjunto de n-cadenas $\mathcal{A}_n$ es un espacio vectorial sobre $\mathbb{F}_q$ de dimensión n, que identificamos naturalmente con

$$\mathbb{F}_q^n = \{(x_1, \ldots, x_n) \mid x_i \in \mathbb{F}_q\}.$$

Proposición 15. *Si $\mathbb{F}_q$ es el cuerpo finito con q elementos, el par $(\mathbb{F}_q^n, d)$ es un espacio métrico*

Demostración. Es necesario comprobar lo siguiente:

(D0) $\mathbb{F}_q^n$ `(espacio vectorial)`

(D1) $d(x,y) \geq 0$ y la $d(x,y) = 0$ si y sólo si x = y, `(positiva definida)`

(D2) $d(x,y) = d(y,x)$, `(simetría)`

(D3) $d(x,z) \leq d(x,y) + d(y,z)$. `(desigualdad triangular)`

Sea $\mathbb{F}_q^n$ un cuerpo finito de q elementos. Si definimos en K^n las operaciones

$$(x_1, \ldots, x_n) + (y_1, \ldots, y_n) = (x_1 + y_1, \ldots, x_n + y_n)$$

$$\lambda(x_1, \ldots, x_n) = (\lambda x_1, \ldots, \lambda x_n)$$

se obtiene un espacio vectorial sobre el cuerpo $\mathbb{F}_q^n$ de dimensión n.

Las propiedades $(D1)$ y $(D2)$ son obvias.

Veamos $(D3)$. Sean $x = x_1 \cdots x_n$, $y = y_1 \cdots y_n$ y $z = z_1 \cdots z_n$. Sea $T = \{i : x_i \neq z_i\}$. Luego $d(x,z) = |T|$.

Como T es la unión disjunta de los conjuntos

$$U = \{i : x_i \neq z_i \text{ y } x_i = y_i\}$$

y

$$V = \{i : x_i \neq z_i \text{ y } x_i \neq y_i\}$$

se tiene que $d(x,z) = |U| + |V|$.

Ahora, por la definición de $d(x,y)$ es inmediato que $|V| \leq d(x,y)$.

Por otra parte, si $i \in U$ entonces $y_i = x_i \neq z_i$ y por lo tanto $|U| \leq d(y,z)$.

Luego d es una distancia. □

Definición 60. Sea un código C, se define la **distancia** de C, y se denota por $d(C)$ ó d_C, como la menor distancia no-nula entre sus palabras.

$$d = d_C = \min_{\substack{x \neq y \\ x,y \in C}} \{d(x,y)\}.$$

Un (n, M, d)-código es un código de *longitud* n, *tamaño* M y *distancia* d.

Definición 61. Sea $x \in \mathbb{F}_q^n$, se define el **peso** de x y se denota por $p(x)$, como el número de coordenadas no-nulas de x.

$$p(x) = \sharp\{1 \leq i \leq n \ : \ x_i \neq 0\}.$$

De otro modo, el peso de x es la distancia de x al $0 = 00 \cdots 0$,

$$p(x) = d(x, 0).$$

Ejemplo 8. El peso de $0120211 \in \mathbb{Z}_3^7$ es $p(0120211) = 5$.

Definición 62. Sea C es un código sobre $\mathbb{F}_q^n$, el peso de C se define por

$$p(C) = \min\{p(x) \mid 0 \neq x \in C\}.$$

Proposición 16. *Sea $x, y \in \mathbb{F}_q^n$ se cumple*

$$d(x,y) = p(x - y).$$

3.3. Códigos lineales.

Definición 63. Un **código lineal q-ario** de longitud n y rango k es un subespacio vectorial $L \subset \mathbb{F}_q^n$ de dimensión k. Así

- el subespacio L es un $[n,k]_q$-código;
- Si L tiene distancia d entonces L es un $[n,k,d]_q$-código;
- Si $L \subset \mathbb{Z}_2^n$, en general se escribe simplemente $[n,k,d]$-código.

Nota 10. El tamaño (número de elementos) de un $[n,k]_q$-código C es $M = q^k$, pues $C \simeq \mathbb{F}_q^k$.

En este caso, la tasa de información es

$$R = \frac{log_q(q^k)}{n} = \frac{k}{n}.$$

Ejemplo 9. El **código de repetición q-ario**

$$Rep_q(n) = \{x \underbrace{\ldots}_{n-2}, x\} \mid x \in \mathbb{F}_q\}$$

con $q < n$ es un $[n,1,n]$-código lineal.

Ejemplo 10. Los códigos C_1, C_3, C_4 y C_6 del Ejemplo 7 son lineales, mientras que C_2 y C_5 no lo son, ya que por ejemplo $01 + 10 = 11 \notin C_2$ y $010 + 012 = 022 \notin C_5$. Además, $C_1 = Z_2$, $C_3 = Rep_2(3)$ y $C_6 = Rep_3(5)$.

Ejemplo 11. El conjunto de todas las palabras de peso par en $\mathbb{F}_2^n$,

$$E(n) = \{x \in \mathbb{F}_2^n \mid p(x) \equiv 0(\text{ mód } 2)\}$$

es un código lineal binario, llamado **código de paridad** con parámetros $[n, n-1, 2]$.

Para calcular la distancia mínima de un (n,M)-código hace falta calcular $\begin{pmatrix} M \\ 2 \end{pmatrix} = \frac{M(M-1)}{2}$ distancias de Hamming. Gracias a la proposición este número se reduce al cálculo de $M-1$ pesos.

Proposición 17. *Si C es un código lineal entonces*

$$d(C) = p(C).$$

Demostración. Como C es lineal, tenemos

$$\begin{aligned} d(C) &= \text{mín}_{x \neq y \in C} d(x,y) = \\ &= \text{mín}_{x \neq y \in C} p(x-y) = \\ &= \text{mín}_{0 \neq x \in C} p(x) = p(C). \end{aligned}$$

ya que $x - y$ recorre todos las palabras código de C cuando x e y recorren todo C. □

Ejemplo 12. El código

$$C = \{000, 011, 101, 110\}$$

es lineal y por lo tanto $d_C = p_C = 2$.

Para el código $C_0 = \{11, 12, 21, 22\}$ se tiene que $d(C_0) = 1 < 2 = p(C_0)$.

Y para el código $C_{00} = \{01, 10\}$ se tiene que $d(C_{00}) = 2 > 1 = p(C_{00})$.

Estos dos últimos códigos no son lineales.

Observación 1. Sea C un (n,M,d)-código.

(1) Los números n, M y d, son los parámetros básicos de C.

(2) La tasa de información R y el número $\delta = d/n$, son parámetros secundarios e influyen en la eficiencia de C durante la transmisión de mensajes.

(3) En general, fijado un n, interesan códigos con:

  (i) M grande (para transmitir muchos mensajes distintos)

  (ii) d grande (para que detecte y corrija el mayor número de errores).

Como son contradictorios entre sí, se busca un balance entre ellos. Por otro lado, existen cotas para los parámetros y parte de la teoría es buscar códigos óptimos que alcancen dichas cotas.

Definición 64. Sean C y C' dos códigos lineales, se dice que son **equivalentes** si uno puede obtenerse del otro por una combinación de operaciones del tipo:

i. Permutación de posiciones;

ii. Permutación de símbolos en una posición fija.

Ejemplo 13. Los códigos

$$C = \{0000, 0101, 1010, 1111\}$$

y

$$C' = \{0000, 0011, 1100, 1111\}$$

son equivalente porque se obtiene uno del otro permutando las posiciones 2 y 3.

Definición 65. Dos (n, M)-códigos q-arios C y C' son equivalentes, y se denota por $C \simeq C'$, si existe una permutación $\sigma \subset S_n$ de las n coordenadas y permutaciones $\pi_1, \ldots, \pi_n \in Biy(\mathcal{A})$ del alfabeto, tales que

$$\begin{aligned} & c_1 c_2 \cdots c_n \in C \Leftrightarrow \\ \Leftrightarrow \quad & \pi_1(c_{\sigma(1)})\pi_2(c_{\sigma(2)}) \cdots \pi_n(c_{\sigma(n)}) \in C' \end{aligned}$$

Observación 2. Si $C \simeq C'$, entonces

$$(n, M, d) = (n', M', d')$$

y por tanto detecta, corrige y tienen la misma tasa de información.

Definición 66. Sean $C \in \mathbb{F}_q^n$ un código y $c \in C$ una palabra codificada. Sea además $e \in \mathbb{F}_q^n$ tal que $p(e) = t$, si al transmitir c se recibe $c' = c + e$, se dice que e es el **patrón de error** y que se ha cometido un error de peso t.

Definición 67. Sea C un código, se dice que C **detecta** t **errores** si cuando en una palabra código se comete un error de peso r, con $1 \leq r \leq t$, la palabra resultante no es una palabra código.

Definición 68. Sea C un código, se dice que C es un **t-detector** si detecta t errores pero no detecta $t + 1$ errores.

Nota 11. Sea $C \subset \mathbb{F}_q^n$ un código con distancia mínima $d(C) = d$, se dice que se decodifica por distancia mínima si para cada palabra recibida se decodifica con la palabra código "más cercana".

Definición 69. Sea C un código, se dice que C **corrige t errores** si, al decodificar por distancia mínima, se pueden corregir todos los errores de peso t o menos.

Definición 70. Sea C un código, se dice que C es **t-corrector** si corrige t errores pero no corrige $(t + 1)$-errores.

Ejemplo 14. El código $C_1 = \{00, 01, 10, 11\}$ no detecta ningún error. El

$$C_2 = \{000, 011, 101, 110\}$$

es un 1-detector pero no corrige. Y

$$C_3 = \{000000, 000111, 111000, 111111\}$$

es un código 2-detector y también 1-corrector.

Proposición 18. *Sea $C \subset \mathbb{F}_q^n$ un código con distancia mínima d. Entonces C es t-detector si y sólo si $d = t + 1$.*

Demostración. Sea $c \in C$ y

$$B(c, r) = \{x \in \mathbb{F}_q^n \mid d(x, c) \leq r\}.$$

Un código detecta errores de peso menor o igual a r si y sólo si $B(c, r)$ no contiene ninguna palabra codificada salvo c; es decir, si y sólo si $r < d$. Por otro lado, si $d(c, c') = d$, y $c, c' \in C$ entonces $c + e = c'$ con $p(e) = d$ y C no detecta este error. □

Proposición 19. *Sea $C \subset \mathbb{F}_q^n$ un código con distancia mínima d. Entonces C es un t-corrector si y sólo si $d = 2t + 1$ ó $d = 2t + 2$.*

Demostración. Que la distancia sea $d = 2t + 1$ ó $d = 2t + 2$ quiere decir que las bolas $B(c, t)$ con $c \in C$ son disjuntas entre sí. Por tanto, C corrige errores de peso t si y sólo si $d \geq 2t + 1$ ó $d \geq 2t + 2$. Además, si para $c \in C$ se tiene que $c + e = c' \in C$ con $p(e) = t + 1$, las bolas $B(c, t + 1) \cap B(c', t + 1) \neq \emptyset$ y por tanto, al cometer un error de peso $t + 1$ en el envío de c, no se puede elegir cuál de las dos palabras del código c o c' es la enviada. □

Corolario 5. *Si un código C tiene distancia mínima d, entonces C es $(d - 1)$-detector y $[\frac{d-1}{2}]$-corrector.*

Cuando se quiere usar un mismo código para detectar y corregir errores de manera simultánea; es decir con el mismo parámetro, y se pretende maximizar las propiedades de corrección, se pierde un poco en la detección de errores.

Teorema 11. *Un código C es simultáneamente t-corrector y $(t + s)$-detector si y sólo si $d = 2t + s + 1$.*

La idea es que, si d es par ($s = 1$), entonces C detecta hasta t errores (aunque en realidad puede detectar más) y corrige hasta $t + 1$ que es una cota mejor que $[\frac{d-1}{2}]$. Si $s = 0$; es decir, d es impar, se tiene el mismo resultado que en el corolario para la corrección.

3.3.1. Matrices generadoras y de paridad.

Definición 71. Sea L un $[n, k]_q$-código. Una **matriz generadora** de L es una matriz $G \in \mathrm{Mat}_{n\times k}(\mathbb{F}_q)$ cuyas columnas forman una base de L.

Ejemplo 15. Sea $G = \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{pmatrix} \in \mathrm{Mat}_{3\times 2}(\mathbb{Z}_2)$. Como las columnas son linealmente independientes, G tiene rango 2 y genera un código binario L con parámetros $[3, 2]$. Además

$$\begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_1 + x_2 \\ x_2 \end{pmatrix}$$

de modo que

$$00 \to 000 \quad 01 \to 011 \quad 10 \to 110 \quad 11 \to 101.$$

Luego $L = \{000, 011, 101, 110\} = E(3)$ y la distancia es $d = 2$.

Sea L un código lineal y sea G una matriz generadora de L. Considerando la aplicación lineal $R_G : \mathbb{F}_q^k \longrightarrow \mathbb{F}_q^n$ definida por $\mathbb{F}_q^k \ni x \to Gx^\perp \in \mathbb{F}_q^n$. La aplicación lineal R_G es un isomorfismo sobre la imagen y por tanto $\operatorname{Im} R_G \simeq L \simeq \mathbb{F}_q^k$. Esto quiere decir que la información se guarda en k coordenadas y el resto $n - k$ son redundantes.

Ejemplo 16. Para el ejemplo anterior es claro que la última coordenada es la de paridad y la información de los mensajes originales está en las dos primeras.

Definición 72. Un $[n, k]$-código q-ario es **sistemático** si existen k coordenadas $i_1, \ldots, i_k$ tal que al restringir las palabras código a estas coordenadas se obtienen todas las q^k palabras de longitud k.

Ejemplo 17. El código

$$C = \{000, 011, 101, 110\}$$

es sistemático en las coordenadas 1 y 2. En realidad, C es sistemático en cualquier par de coordenadas.

Si G genera L, entonces toda matriz equivalente en la que se realizan operaciones elementales por columnas es también una matriz generadora del mismo código, ya que sólo cambia la base de L.

Ejemplo 18. La matriz

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

genera el código dado por las ecuaciones $(x_1 + x_3, x_1 + x_2, x_2, x_1 + x_2 + x_3)$. Haciendo operaciones elementales por columnas en G se obtiene la matriz

$$G' = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \end{pmatrix}$$

y el mismo código tiene las ecuaciones más sencillas $(x_1, x_2, x_3, x_1 + x_3)$. Éste es

$$\begin{aligned} L \quad = \quad & \{0000, 0011, 0100, 0111, 1001, \\ & 1010, 1100, 1111\}. \end{aligned}$$

Con esta forma de expresar el código se ve que es sistemático en las tres primeras coordenadas.

Proposición 20. *Todo $[n, k]$-código lineal L es sistemático en k coordenadas.*

Demostración. Basta reducir una matriz generadora de L a una matriz que sea la matriz de la forma $\begin{pmatrix} Id_k \\ \hline A \end{pmatrix}$ donde Id_k es la matriz identidad $k \times k$ y $A \in \operatorname{Mat}_{n-k \times k}$. □

Definición 73. Sea G una matriz generadora de un código lineal L. Se dice que G está en forma **estándar** si es de la forma $G = \begin{pmatrix} Id_k \\ \hline A \end{pmatrix}$ donde Id_k es la matriz identidad $k \times k$ y $A \in \operatorname{Mat}_{n-k \times k}$.

Si G está en forma estándar, entonces L es sistemático en las k primeras coordenadas pues $Gx^\perp = \begin{pmatrix} Id_k \\ A \end{pmatrix} x^\perp = \begin{pmatrix} x^\perp \\ Ax^\perp \end{pmatrix}$.

En esta situación, codificar y decodificar es trivial ya que el esquema resulta

$$x^\perp \underset{cod}{\rightarrow} Gx^\perp = \begin{pmatrix} x^\perp \\ Ax^\perp \end{pmatrix} \underset{dec}{\rightarrow} x^\perp.$$

Por otra parte, no todo código lineal tiene matriz generadora en forma estándar. Por ejemplo, si L es el código generado por

$$G = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix}.$$

Sin embargo, se tiene el siguiente resultado:

Proposición 21. *Todo código lineal L es equivalente a un código L' cuya matriz generadora está en forma estándar.*

Corolario 6. *Dado un $[n,k]$-código L y $1 \leq i_1 \leq \cdots \leq i_k \leq n$, existe un código lineal L' equivalente a L, y sistemático en las coordenadas $i_1, \ldots, i_k$.*

Definición 74. El espacio vectorial $\mathbb{F}_q^n$ tiene un producto interno natural dado por

$$x \cdot y = x_1 y_1 + \cdots + x_n y_n \quad x, y \in \mathbb{F}_q^n.$$

Definición 75. Si L es un $[n,k]_q$-código, el conjunto

$$L^\perp = \{x \in \mathbb{F}_q^n \mid x \cdot c = 0 \text{ para todo } c \in L\}$$

se denomina **código dual** de L.

Teorema 12. *Sea L es un $[n,k]_q$-código.*

(1) Si G es una matriz generadora de L entonces

$$\begin{aligned} L^\perp &= \{x \in \mathbb{F}_q^n \mid G^\perp x^\perp = 0\} = \\ &= \{x \in \mathbb{F}_q^n \mid xG = 0\} \end{aligned}$$

(2) $L^\perp$ es un $[n, n-k]_q$-código.

(3) ${L^\perp}^\perp = L$.

Demostración. (1) Por definición, $x \in L^\perp$ si y sólo si $x \cdot c = 0$ para todo $c \in L$. Luego,

$$0 = x \cdot c = xc^\perp = x(Gu^\perp) = (xG)u^\perp$$

para todo $u \in \mathbb{F}_q^k$. Si $xG = 0$ entonces $x \in L^\perp$.

Recíprocamente, si $x \in L^\perp$ entonces $(xG)u^\perp = 0$ para todo $u \in \mathbb{F}_q^k$. En particular, para $u = e_1, \ldots, e_k$, los vectores de la base canónica, se tiene que $0 = (G^\perp x^\perp)e_i^\perp = (xG)^i$ para $1 \leq i \leq k$. Por lo tanto $xG = 0$.

(2) Es claro que $L^\perp$ es un subespacio de $\mathbb{F}_q^n$. Por (i), $L^\perp = \{x \in \mathbb{F}_q^n \mid xG = 0\}$ o sea $L^\perp$ es el espacio solución de k ecuaciones con n incógnitas. Luego, como G tiene rango k, hay $n-k$ variables libres, por lo tanto $\dim L^\perp = n - k$.

(3) Se tiene que

$$L \subset (L^\perp)^\perp = \{x \in \mathbb{F}_q^n \mid x \cdot c' = 0\}$$

para todo $c' \in L^\perp$. Pero

$$\dim(L^\perp)^\perp = n - (n-k) = k = \dim L,$$

luego $L = (L^\perp)^\perp$. □

Definición 76. Sea L un $[n,k]_q$-código. Una matriz $H \in \mathrm{Mat}_{n-k\times n}(\mathbb{F}_q)$ se dice **matriz de paridad** de L si es una matriz generadora *por filas* de $L^\perp$ (es decir, las columnas de $H^\perp$ generan $L^\perp$).

Observación 3. (1) H siempre existe y $H \in \mathrm{Mat}_{n-k\times n}(\mathbb{F}_q)$.

(2) Se cumple $HG = 0$. En efecto, sean $c \in L$ y $c' \in L^\perp$, entonces $c = Gu^\perp$ y $c' = H^\perp w^\perp$ para ciertos $u \in \mathbb{F}_q^k, w \in \mathbb{F}_q^{n-k}$. Luego, $c \cdot c' = 0$ si y sólo si

$$0 = H^\perp w^\perp \cdot Gu^\perp = wHGu^\perp$$

lo que a su vez sucede si y sólo si $HG = 0$, pues $e_i(HG)e_j^\perp = (HG)_{ij}$.

(3) Si $H \in \mathrm{Mat}_{n-k\times n}(\mathbb{F}_q)$ y $HG = 0$ entonces H genera $L^\perp$, por lo tanto es una matriz de paridad de L.

(4) Si G es una matriz generadora de L entonces $G^\perp$ es una matriz de paridad de $L^\perp$. Esto es así pues la matriz de paridad de $L^\perp$ es $G^\perp$ porque $(G^\perp)^\perp = G$ es la matriz generadora de $(L^\perp)^\perp = L$.

Proposición 22. *Sea H la matriz de paridad de un $[n,k]_q$-código L. Entonces,*

$$\begin{aligned} L &= \{x \in \mathbb{F}_q^n \mid Hx^\perp = 0\} = \\ &= \{x \in \mathbb{F}_q^n \mid xH^\perp = 0\}. \end{aligned}$$

Demostración. Si $c \in L$, entonces $c = Gx^\perp$ donde $x \in \mathbb{F}_q^k$ y G es la matriz generadora de L. Luego $Hc^\perp = HGx^\perp = 0$ y por lo tanto $L \subset S_H = \{x \in \mathbb{F}_q^n \mid Hx^\perp = 0\}$, donde S_H es el espacio solución de un sistema de $n-k$ ecuaciones con n incógnitas y rango $n-k$. Como $\dim S_H = n-(n-k) = k = \dim L$, tenemos que $L = \{x \in \mathbb{F}_q^n \mid Hx^\perp = 0\}$. □

Para obtener H a partir del código L, se buscan las ecuaciones implícitas del código. Una forma rápida es utilizar la matriz generadora G.

Si $G = \left(\begin{array}{c} Id_k \\ \hline A \end{array}\right)$ está en forma estándar, entonces $H = \left(\begin{array}{cc} -A & |Id_{n-k} \end{array}\right)$ es una matriz de paridad de L. Pues

$$HG = \left(\begin{array}{cc} -A & |Id_{n-k} \end{array}\right)\left(\begin{array}{c} Id_k \\ \hline A \end{array}\right) = -A + A = 0$$

Definición 77. Sea H la matriz de paridad de un código L. Si $H = \left(\begin{array}{cc} -A & |Id_{n-k} \end{array}\right)$ se le llama **matriz de paridad en forma estándar** de L. (Aunque no está en forma estándar para $L^\perp$.)

Definición 78. Un código lineal L se dice **auto-ortogonal** si $L \subset L^\perp$ y **autodual** si $L = L^\perp$.

Un código autodual tiene parámetros $[2m, m]$. Si L es un código autodual, toda matriz generadora de L es matriz de paridad y recíprocamente. Luego, si $G = \left(\begin{array}{c} Id_m \\ \hline A \end{array}\right)$ es una matriz generadora de L, entonces $H^t = \left(\begin{array}{c} -A \\ \hline Id_m \end{array}\right)$ también lo es.

Teorema 13. *Sea L un $[n,k,d]_q$-código y H una matriz de paridad de L. Entonces H tiene d columnas linealmente dependientes, pero cualquier conjunto de $d-1$ columnas son linealmente independientes.*

$$d = \textit{mín}\,\{r \mid \textit{hay } r \textit{ columnas l. d. en } H\}.$$

Demostración. Sean $H_1, \ldots, H_n$ las columnas de H.

$$\begin{aligned} c \quad &\in \quad L \subset \mathbb{F}_q^n \Leftrightarrow Hc^{\perp} = 0 \Leftrightarrow \\ &\Leftrightarrow \quad H^1 c_1 + \cdots + H^n c_n = 0 \end{aligned}$$

Pero si $c \in L$, c tiene peso r si y sólo si hay un conjunto de r columnas linealmente dependiente en H. Como $r \geq d$, no puede haber $d-1$ columnas linealmente dependientes en H. □

Este teorema se puede usar para construir códigos lineales con distancia d prefijada.

Proposición 23 (Singleton)**.** *Si L es un $[n,k,d]_q$-código entonces*

$$d \leq n - k + 1.$$

Demostración. La matriz de paridad H pertenece al espacio vectorial $\mathrm{Mat}\, n-k \times n(\mathbb{F}_q)$ y por el teorema anterior cualquier conjunto de $d-1$ columnas de H son linealmente independientes, luego $d-1 \leq n-k$. □

Ejemplo 19. Un $[11,6,d]_q$-código tiene $d \leq 11-6+1 = 6$.

Un $[11,k,7]_q$-código tiene $k \geq n-d+1 = 5$.

Un $[n,8,7]_q$-código tiene $n \geq d+k-1 = 14$.

No existen $[n, n-1, 3]_q$-códigos.

Observación 4. Los códigos cuyos parámetros alcanzan la igualdad en la cota de Singleton se llaman MDS (por "maximum distance separable"), ya que tienen la mayor distancia posible, dados una longitud y un tamaño fijos.

Cuando el código $L \subset \mathbb{F}_q^n$ es lineal, se puede utilizar la definición de espacio cociente

$$\mathbb{F}_q^n / L = \{x + L \mid x \in \mathbb{F}_q^n\}$$

para decodificar. El número de clases es

$$|\mathbb{F}_q^n / L| = |\mathbb{F}_q^n| / |L| = q^{n-k}.$$

Definición 79. Sea L un $[n,k]_q$-código con matriz de paridad H. Si $x \in \mathbb{F}_q^n$, el **síndrome de x** se define por

$$s(x) = s_H(x) := Hx^{\perp}.$$

Observación 5. Se tiene por definición que $x \in L$ si y sólo si $s(x) = 0$.

Y, x e y tienen el mismo síndrome si y sólo si x, y están en la misma clase módulo L. En efecto,

$$\begin{aligned} x + L = y + L \quad &\Leftrightarrow \quad x - y \in L \Leftrightarrow H(x-y) = 0 \Leftrightarrow \\ &\Leftrightarrow \quad Hx = Hy \end{aligned}$$

De esto modo, para buscar la función decodificadora usando síndromes por distancia mínima, se hace lo siguiente:

Sea $x \in \mathbb{F}_q^n$ una palabra enviada. Se busca la palabra código a distancia mínima de x. Supóngase que

$$s = d(x,c) = \mathrm{mín}_{c' \in L} d(x,c').$$

Así, se puede escribir $x = c + e$ con $c \in L$ y $e \notin L$ una palabra con peso mínimo s. Como L es lineal

$$\begin{aligned} &\mathrm{mín}_{c \in L} d(x,c) = \ \mathrm{mín}_{c \in L} p(x-c) = \\ = \quad &\mathrm{mín}_{e \in x+L} p(e), \end{aligned}$$

por tanto, tratar de decodificar x como una palabra del código c a distancia mínima s es equivalente a buscar $e = x - c$ con peso mínimo en $x + L$.

Teorema 14. *Sea L un código lineal con matriz de paridad H. Decodificar la palabra recibida x como $x - e = c \in L$ por distancia mínima es equivalente a encontrar e con peso mínimo en la clase $x + L$, es decir, donde e es una palabra de peso mínimo con igual síndrome que x.*

Ejemplo 20. Sea L el $[4, 2]$-código binario dado por la matriz generadora

$$G = \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 0 \\ 1 & 0 \end{pmatrix}.$$

Las clases son

$$\begin{array}{rl} 0000 + L = & \{0000, 0100, 1101, 1001\} \\ 1000 + L = & \{1000, 1100, 0101, 0001\} \\ 0010 + L = & \{0010, 0110, 1111, 1011\} \\ 1010 + L = & \{1010, 1110, 0111, 0011\} \end{array}.$$

Elegimos los representantes de las clases de peso mínimo y así el arreglo es

$$\begin{array}{cccc} 0000 & 0100 & 1101 & 1001 \\ 1000 & 1100 & 0101 & 0001 \\ 0010 & 0110 & 1111 & 1011 \\ 1010 & 1110 & 0111 & 0011 \end{array}$$

Si se recibe la palabra $x = 0111$, se detecta que hay un error pues x no está en la primer fila del arreglo. Luego, ésta es corregida como la palabra código $c = 1101$, que está arriba en la misma columna que x.

Sólo es necesario almacenar una tabla con los representantes de las clases de peso mínimo y sus correspondientes síndromes. Si se recibe la palabra x, se calcula su síndrome $s(x)$ y se busca en la tabla el representante de la clase a_i con igual síndrome que x. Luego, se decodifica x como $c = x - a_i$. Esta es la llamada *decodificación por síndrome.*

3.3.2. Códigos de Hamming.

Ejemplo 21. Sea

$$H = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix} \in \mathrm{Mat}_{3\times 7}(\mathbb{F}_2)$$

cuyas fila son las $2^3 - 1 = 7$ palabras no-nulas de $\mathbb{F}_2^3$, escritas en forma ascendente. Si se considera que H es la matriz de paridad de un código lineal C con parámetros $[7, 4, d]$, se puede calcular la distancia mínima de C usando el Teorema 13. Como cualquier par de columnas de H son linealmente independientes, pero H tiene 3 columnas linealmente dependientes, entonces C tiene distancia $d = 3$. Para encontrar las $2^4 = 16$ palabras código de C sólo tenemos que resolver las ecuaciones de paridad determinadas por H

$$\begin{array}{rcl} x_4 + x_5 + x_6 + x_7 & = & 0 \\ x_2 + x_3 + x_6 + x_7 & = & 0 \\ x_1 + x_3 + x_5 + x_7 & = & 0. \end{array}$$

Una base de C se obtiene tomando a x_3, x_5, x_6 y x_7 como variables libres. Es decir $c_1 = 1110000, c_2 = 1101100, c_3 = 0101010$ y $c_4 = 11010001$. Luego, el código C se obtiene haciendo todas las sumas posibles entre c_1, c_2, c_3 y c_4. Este es el código de Hamming binario de longitud 7, denotado por $\mathcal{H}_2(3)$.

La construcción anterior se puede hacer para cualquier $r \geq 2$. Es decir, si formamos la matriz H cuyas columnas son las $2^r - 1$ palabras no-nulas de $\mathbb{F}_2^r$, tenemos una matriz $r \times n$, con $n = 2^r - 1$, en donde cualquier par de columnas son linealmente independientes, pero hay 3 columnas linealmente dependientes. Luego, H es la matriz de paridad de un código lineal denotado por $\mathcal{H}_2(r)$ con parámetros

$$[n = 2^r - 1, k = n - r = 2^r - r - 1, d = 3].$$

Este es el llamado **código de Hamming binario de orden** r.

Por ejemplo, $\mathcal{H}_2(4)$ tiene parámetros $[15, 11, 3]$, luego codifica $2^{11} = 2048$ mensajes y corrige 1 error.

Utilizando el procedimiento anterior, los códigos de Hamming binarios pueden generalizarse a cualquier alfabeto $\mathbb{F}_q$. Para cada r, queremos construir una matriz $\mathcal{H}_{q,r} \in \mathrm{Mat}_{r\times n}(\mathbb{F}_q)$, con el mayor número de columnas, de modo que cualquier par de columnas sean linealmente independientes (o sea ninguna columna es múltiplo de otra), pero que algún conjunto de tres columnas sea linealmente dependiente. Para cada r fijo, construimos la matriz $\mathcal{H}_{q,r}$ de la siguiente manera. Elegimos cualquier columna no-nula $c_1 \in V_1 = \mathbb{F}_q^r$. Luego elegimos cualquier columna no-nula

$$c_2 \in V_2 = V_1 - \{\alpha c_1 \mid \alpha \in \mathbb{F}_q^*\}.$$

Continuamos eligiendo columnas no-nulas de esta forma y descartamos los múltiplos escalares de las columnas elegidas hasta agotar todas las columnas de $\mathbb{F}_q^r$. Como cada columna $c \in \mathbb{F}_q^r$ tiene $q - 1$ múltiplos escalares no-nulos αc, $\alpha \in \mathbb{F}_q^*$, vemos que la matriz $H_{q,r}$ formada por las columnas c_i elegidas como antes tiene $(q^r - 1)/(q - 1)$ columnas.

Definición 80. La matriz $H_{q,r} \in \mathrm{Mat}_{r\times n}(\mathbb{F}_q)$, con $n = (q^r - 1)/(q - 1)$, se llama matriz de Hamming de orden r y es la matriz de paridad de un código lineal q-ario con parámetros

$$n = \frac{q^r - 1}{q - 1},\ k = n - r,\ d = 3$$

denotado por $\mathcal{H}_q(r)$ y llamado código de Hamming q-ario de orden r.

Proposición 24. *Los códigos de Hamming $\mathcal{H}_q(r)$ tienen parámetros $(\frac{q^r-1}{q-1}, q^{n-r}, 3)$ y por tanto corrigen 1 error.*

La decodificación por síndrome con algunas matrices de Hamming es más elegante que con otras. Las matrices en donde las columnas son la trascripción de los números naturales en el cuerpo $\mathbb{F}_q$ (primera coordenada no-nula igual a 1) son un ejemplo de esto. En el caso binario, si cometemos un error en la transmisión en la i-ésima coordenada, el vector error que resulta es e_i. Luego, el síndrome de la palabra recibida es $He_i^\perp$, que es la transpuesta de la i-ésima columna H_i de H. Más aún, el número binario que representa este vector coincide con la posición del error, es decir i.

Ejemplo 22. Sea $\mathcal{H}_2(3)$ el código de Hamming definido por la matriz

$$H_{3,2} = H = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix} \in \mathrm{Mat}_{3\times 7}(\mathbb{F}_2).$$

Supongamos que cometemos un error en la tercer coordenada, es decir el patrón de error es $e_3 = 0010000$. Luego, $He_3^\perp = H(0010000)^\perp = H_3 = (011)^\perp = (011)_2 = 3_{10}$. Es decir, el síndrome determina la coordenada errónea.

El caso general, para $H_q(r)$, es muy parecido. Si cometemos un error en la coordenada i, el vector error es de la forma αe_i, con $\alpha \in \mathbb{F}_q$. Luego, el síndrome es $s(e_i) = H_{q,r}(\alpha e_i)^\perp$ que es α multiplicado por la i-ésima columna de $H_{q,r}$. Por la forma en que construimos H, vemos que α es la primera coordenada no-nula de $s(e_i)$. Multiplicando $s(e_i)$ por α^{-1} obtenemos la columna i de H, lo cual nos da la coordenada del error.

Ejemplo 23. Sea $\mathcal{H}_3(3)$ el código de Hamming definido por la matriz

$$H = H_{3,3} = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 2 & 2 & 2 \\ 1 & 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 \end{pmatrix}$$

con parámetros $[13, 10, 3]$ Si recibimos la palabra $x = 1101112211201$, su síndrome es

$$Hx^{\perp} = (201)^{\perp} = 2(102) = 2 \times H_7 \quad \text{(columna 7 de } H\text{)}$$

Por lo tanto, detectamos que hay un error de magnitud 2 en la coordenada 7 de la palabra recibida x. Luego, decodificamos x como

$$c = x - 2e_7 = 1101112211201 - 00000020000 = 1101110211201$$

Capítulo 4

Cálculo numérico

Métodos iterativos para resolución de funciones no lineales

Beatriz Graña Otero

Usal, Salamanca

ITIS, 27 de Noviembre de 2009

Outline

Los métodos iterativos buscan aproximaciones

que en nuestro caso serán de las raíces de una función
$f : \mathbb{R} \to \mathbb{R}$

Se debe tener en cuenta:

1. Complejidad del Algoritmo (número de operaciones que se realizan, cálculos necesarios intermedios, evaluaciones)
2. Análisis de errores (número de iteraciones, error, tolerancia)
3. Convergencia del algoritmo

Y su idea es:

1. Iniciar con una estimación motivada por las condiciones al contorno o si falta esta información, a ciegas.
2. Construir una sucesión $\{x_n\} \subset \mathbb{R}$ convergente a una raíz r de f
3. Criterio de parada del algoritmo que viene motivado porque las iteraciones varíen poco $e_n \approx |x_n - x_{n+1}|$, la ecuación se satisfaga con una aproximación suficiente $|f(x_n)|$ o se alcance un número máximo de iteraciones (por si el algoritmo diverge o converge muy lentamente).

Por tanto, se trata de crear iterativamente una sucesión de número reales $\{x_n\}$ convergente a una raíz r de la función f de modo que $\lim_{n\to+\infty} f(x_n) = f(r)$.

Definición

Sea $\{x_n\} \subset \mathbb{R}$ una sucesión de números reales, se dice que es **convergente** *a r si $\forall\ \epsilon > 0$ existe $n_\epsilon \in \mathbb{N}$ tal que $|x_n - r| < \epsilon$ para todo $n > n_\epsilon$.*

Definición

Sea $\{x_n\} \subset \mathbb{R}$ una sucesión de números reales, se dice que es una sucesión de **Cauchy** *si $\forall\ \epsilon > 0$ existe $n_\epsilon \in \mathbb{N}$ tal que $|x_n - x_{n'}| < \epsilon$ para todo $n, n' > n_\epsilon$.*

Se puede demostrar que toda sucesión en $\mathbb{R}$ es convergente si y sólo si es de de Cauchy.

Proposición

Sea $f : \mathbb{R} \to \mathbb{R}$ una función continua, si $\{x_n\}$ es una sucesión convergente a r, entonces

$$f(\lim_{n\to+\infty}(x_n)) = \lim_{n\to+\infty} f(x_n) = f(r)$$

Los métodos que utilizaremos se basan todos en el siguiente teorema.

Teorema (Teorema de Bolzano)

Sea $f : [a, b] \to \mathbb{R}$ un función continua. Si

$$\operatorname{signo} f(a) \neq \operatorname{signo} f(b),$$

entonces f tiene un cero en $[a, b]$.

Análisis de errores

Sea f función con un cero en $[a, b]$. Entonces:

1. Error a priori. Al tomar x_n como solución aproximada de r, el error cometido es $|x_n - r| = e_n$ y se demuestra que si la sucesión converge entonces $e_n \approx |x_n - x_{n-1}|$
2. Error a posteriori. Sea x_n una solución aproximada de f obtenida por algún método. La evaluación $f(x_n) \neq 0$ nos da una magnitud del error cometido, pues su valor es la distancia al cero. Sin embargo, el error depende de cómo sea la función f,

$$e_n = |x_n - r| \approx \frac{|f(x_n)|}{|f'(x)|} \leq \frac{f(x_n)}{\text{mín}_{x\in[a,b]}|f'(x)|}$$

Método de la bisección

Se crea una sucesión $\{x_n\}$ convergente a la raíz $r \in [a, b]$ de f de la siguiente manera:
Para $\forall i, \ldots n, \ldots$, sea el intervalo $I_i = [a_i, b_i]$ y sea $x_i = \frac{a_i + b_i}{2}$ de forma que

$$I_1 = [a, b]$$

e

$$I_i = \begin{cases} [a_i, x_i] & \text{si signo } f(a_i) \neq \text{signo } f(x_i) \\ [x_i, b_i] & \text{si signo } f(x_i) \neq \text{signo } f(b_i) \end{cases}$$

Así

$$x_n = \frac{a_n + b_n}{2}$$

El algoritmo se detiene cuando $f(x_i) < \text{tol}$ ó $|b_i - a_i| < e$ con e y tol prefijados.

Análisis de errores

Sea r la solución de la función f, al dar el término x_n de la sucesión creada por el método de la bisección y si se toma como solución, se está cometiendo un error

$$e_n = |r - x_n| \leq \frac{|b-a|}{2^{(n+1)}}$$

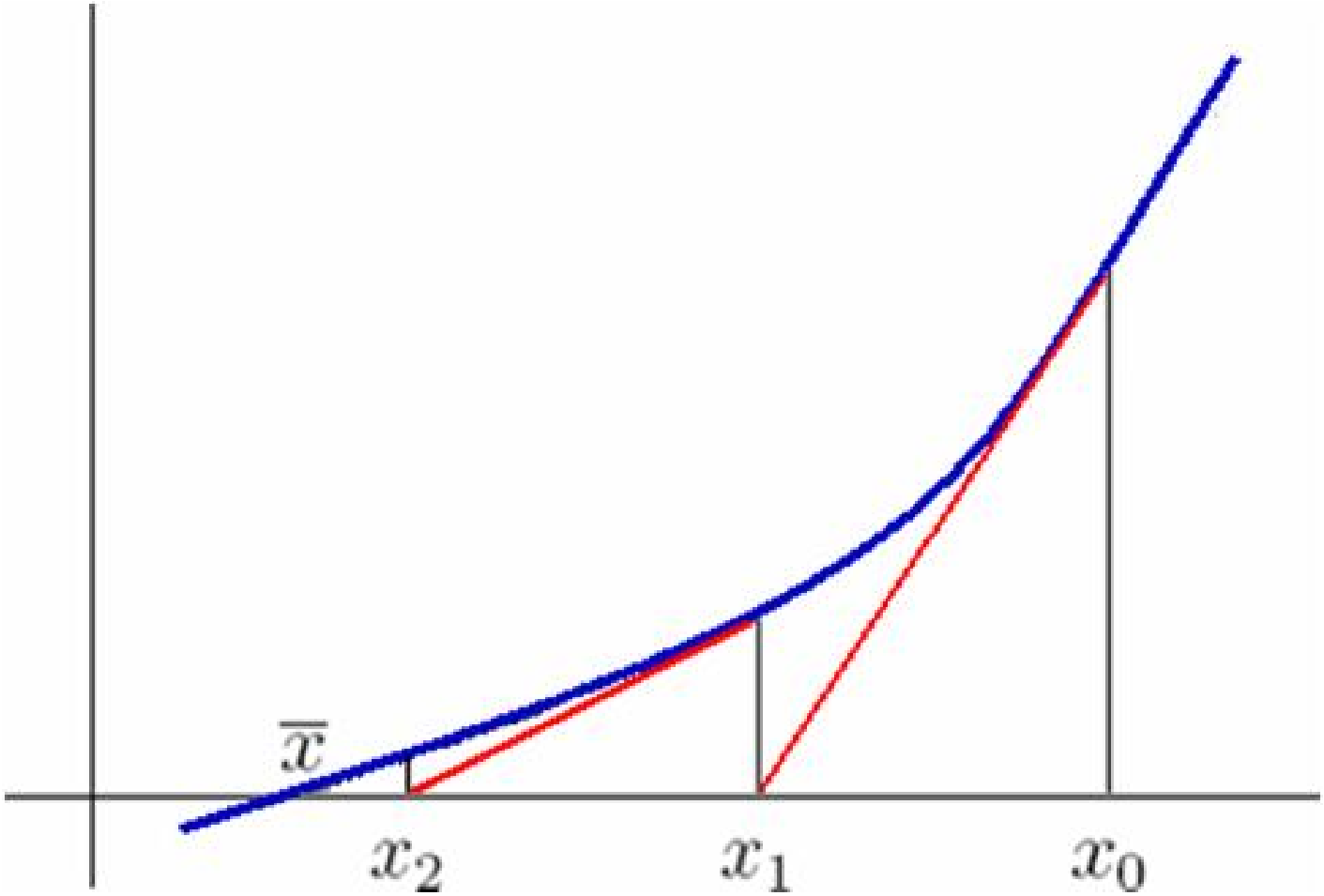
Con n iteraciones la solución aproximada es x_n con un error de $\pm 2^{-(n+1)}|b-a|$.

Ejemplo

Se pretende calcular el número de iteraciones necesarias para obtener una precisión de 6 *decimales. Si* $|b-a| = 1$ *entonces* $2^{-(n+1)}|b-a| \leq 10^{-6}$. *Haciendo los cálculos se obtiene que* $-(n+1)\ln 2 \leq -6 \ln 10$, *luego*

$$n \geq \frac{6 \ln 10}{\ln 2} - 1$$

Método de Newton



Algoritmo

La entrada del algoritmo son, el intervalo $[a, b]$, la variable x, el error e, la función $f(x)$ y la tolerancia tol.
Se estudia un valor inicial x_0 y se calculan los demás elementos de la sucesión con la fórmula

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

El algoritmo se detiene cuando se alcanza o el error o la tolerancia. A veces se introduce otro parámetro que es el número de iteraciones.

Criterios de convergencia

Sea $f : [a, b] \to \mathbb{R}$

1. **continua**, dos veces continuamente **derivable**,
2. $f(a)f(b) < 0$,
3. $f'(x) \neq 0$ en $[a, b]$,
4. $f''(x) \neq 0$ en $[a, b]$.

entonces el método de Newton construye una sucesión $\{x_n\}$ convergente a una raíz $r \in [a, b]$ de f. Además,

$$\begin{cases} x_0 = a & \text{si signo } f(a) = \text{signo } f''(a) \\ x_0 = b & \text{en otro caso} \end{cases}$$

Análisis de errores

Sea r un cero de la función f en $[a, b]$, y sea x_n el elemento de la sucesión al realizar la iteración n del algoritmo, entonces el error cometido es

$$e_n = |x_n - r| \leq \frac{M_2}{2m_1}|x_n - x_{n-1}|^2,$$

donde

$$M_2 = \text{máx}_{x\in[a,b]} |f''(x)|$$

y

$$m_1 = \text{mín}_{x\in[a,b]} |f'(x)|$$

Velocidad de convergencia

El error en el paso $n+1$ es $e_{n+1} = |x_{n+1} - r|$, sea $h_{n+1} = r - x_{n+1}$ entonces, cuando la sucesión se obtiene por el método de Newton resulta que:

$$h_{n+1} = r - x_{n+1} = r - x_n + \frac{f(x_n)}{f'(x_n)} = h_n + \frac{f(x_n)}{f'(x_n)}$$

y

$$0 = f(r) = f(x_n + h_n) = f(x_n) + f'(x_n)h_n + \frac{f''(x_n)}{2}h_n^2 + o(h_n^3)$$

o equivalentemente

$$0 = f(x_n) + f'(x_n)h_n + \frac{f''(t)}{2}h_n^2 \text{ con } t \in B(r, e_n)$$

Por otro lado, si $f'(x) \neq 0$ para todo $x \in [a, b]$, entonces

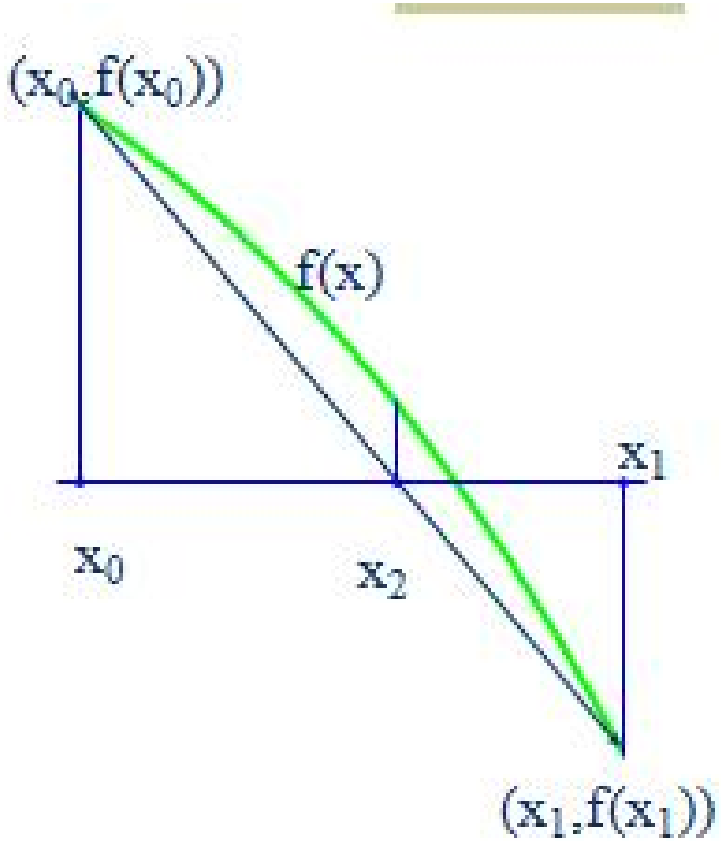
$$0 = \frac{f(x_n)}{f'(x_n)} + h_n + \frac{f''(t)}{2f'(x_n)} h_n^2$$

que tomando valores absolutos resulta ser

$$e_{n+1} = \frac{|f''(t)|}{2|f'(x_n)|} e_n^2 \leq \frac{\text{máx}_{x\in[a,b]}|f''(x)|}{\text{mín}_{x\in[a,b]}2|f'(x)|} e_n^2$$

Por tanto la convergencia es cuadrática $e_{n+1}/e_n^2 \approx cte$..

Método de Secante



Método de la secante

1. Criterios de convergencia iguales al método de Newton
2. Algoritmo dado por la función generadora

$$x_{n+1} = x_n + \frac{f(x_n)}{m}$$

donde $m = \frac{f(x_n)-f(x_{n-1})}{x_n-x_{n-1}}$

3. Análisis de errores similares al método de Newton aunque un poco más lento. La convergencia es superlineal $e_{n+1}/e_n^{1,618} \approx cte$.
4. La ventaja es que se evita el cálculo de la derivada.

Relaciones de equivalencia

1. Una relación de equivalencia $\equiv$ en un conjunto X se puede interpretar como el subconjunto de $X \times X$ dado por $\{(x, x') \in X \times X \mid x \equiv x'\}$. Enúnciesen las propiedades de la relación de equivalencia en términos de dicho subconjunto. Si $R, S \subset X \times X$ representan dos relaciones de equivalencia, ¿ representa $R \cap S$ una relación de equivalencia? ¿ y $R \cup S$?.

2. Sea $A = \{1, 2, 3, 4\}$ y la relación: $aRb \iff a \mid b$. Calcular la matriz de la relación.

3. Sea $A = \{a, b, c, d\}$ y $R \equiv \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$. Determinar $R \subset A \times A$.

4. Sea $A = \{2, 6, 8, 15, 18\}$. Demostrar que las siguientes relaciones son de equivalencia y calcular la matriz asociada a cada una de ellas:

   a) $xRy \iff 6 \mid x - y$
   b) $xSy \iff 4 \mid x - y$

5. Decidir cuáles de las siguientes relaciones son de equivalencia:

   a) En $\mathbb{Z}$, $m \equiv n \iff m \geq n$ (ídem con $m > n,\ mn \geq 0$).
   b) En $\mathbb{N}$, $m \equiv n \iff m + n \geq 0$ (ídem con $m + n > 0,\ 2m \geq n$).
   c) En $\mathbb{R} \times \mathbb{R}$, $(x, y) \equiv (x', y') \iff x < x'$ ó $(x = x'$ e $y < y')$.

6. Sea $f \colon X \longrightarrow Y$ una aplicación de conjuntos. Se define una relación R_f en X por

   $$x \equiv x' (\text{mod } R_f) \text{ si y sólo si } f(x) = f(x')$$

   Probar que R_f es una relación de equivalencia.

7. En $\mathbb{Z} \times (\mathbb{Z} - \{0\})$ se define la relación:

   $$(a, b) \equiv (a', b') \text{ si y sólo si } ab' = a'b$$

   Probar que es de equivalencia. El conjunto cociente se denomina *conjunto de los números racionales.*

8. Sea X el conjunto de $\mathbb{R}^2 - \{(0, 0)\}$. Se define en X la siguiente relación:

   $$x \equiv y \text{ si y sólo si existe una semirrecta que parte del origen y pasa por } x \text{ e } y$$

   Probar que es una relación de equivalencia y que el conjunto cociente se identifica de modo natural con una circunferencia centrada en el $(0, 0)$.

9. Sea X el conjunto de $\mathbb{R}^n - \{0\}$. Se define en X la siguiente relación:

   $$x \equiv y \text{ si y sólo si existe un } \lambda \in \mathbb{R} \text{ tal que } x = \lambda y$$

   Probar que es una relación de equivalencia.

10. En $\mathbb{R}^2$ se define la relación:

    $$(x, y) \equiv (x', y') \text{ si y sólo si } xy = x'y'$$

    Probar que es de equivalencia y calcular las clases de equivalencia.

11. Fijado un número natural n, se define en $\mathbb{Z}$ la relación:

    $$m \equiv m' \pmod{n} \text{ si sólo si } m - m' \text{ es múltiplo de } n$$

    Probar que es de equivalencia calcular las clases de equivalencia. El conjunto cociente se denota por $\mathbb{Z}/n$.

Relaciones de orden

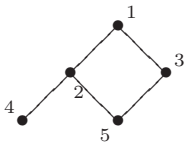
12. Dada la matriz:

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

   a) Hallar el diagrama de Hasse asociado.

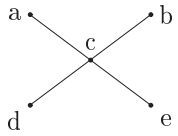
   b) Dar un orden total que contenga a R.

13. Sea $A = \{1, 2, 3, 4, 5\}$ con el orden:



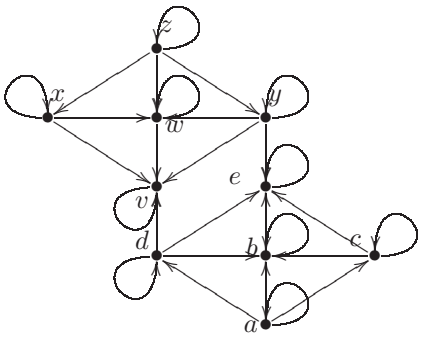
   Sea $B = \{$subconjuntos totalmente ordenados de A con 2 o más elementos$\}$ ordenado con la inclusión de conjuntos. Construir el diagrama de Hasse de B.

14. Sea $A = \{a, b, c, d, e\}$ con el orden dado por:



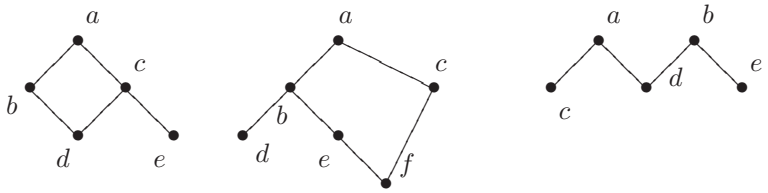
   Determinar si $\{a, c, d\}$, $\{b, d\}$, $\{a, b, c\}$, $\{b, c, e\}$ y $\{d, e\}$ son totalmente ordenados o no. Dar para cada subconjunto los elementos maximales, minimales, supremo e ínfimo.

15. Sea $A = \{a, b, c, d, e, v, w, x, y, z\}$ con el orden dado por:



   Calcular: $\inf\{b, d\}$, $\inf\{b, w\}$, $\sup\{c, x\}$ y $\sup\{d, e\}$.

16. Hallar maximales, minimales, máximo y mínimo de los siguientes conjuntos ordenados según sus diagramas de Hasse y dar una ordenación total para cada uno.



17. Sea $U = \{1, 2, 3, 4\}$ y $A = \mathcal{P}(U)$. Consideramos en A la relación de orden dada por la inclusión de conjuntos. Calcular supremos, ínfimos, máximos y mínimos (si existen) de los siguientes conjuntos:

    a) $B = \{\{1\}, \{2\}\}$

    b) $B = \{\{1\}, \{2\}, \{3\}, \{1, 2\}\}$

    c) $B = \{\emptyset, \{1\}, \{2\}, \{1, 2\}\}$

    d) $B = \{\{1\}, \{1, 2\}, \{1, 2, 3\}\}$

    e) $B = \{\{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}\}$

18. Sea $X = \{0, 1\}$. En $A = X \times X$ definimos la siguiente relación:

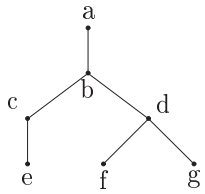
$$(a, b)R(c, d) \iff \left\{ \begin{array}{l} a < c \\ \text{o} \\ a = c \text{ y } b \leq d \end{array} \right\}$$

    a) Demostrar que es un orden parcial. Es total?

    b) Calcular maximales y minimales.

    c) Calcular máximos y mínimos.

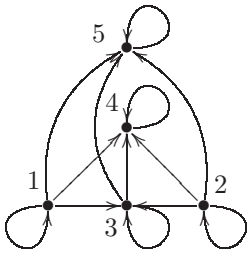
19. Sea $A = \{a, b, c, d, e, f, g\}$ y el orden dado por:



    Calcular maximales, minimales, máximo y mínimo (si existen) de A.

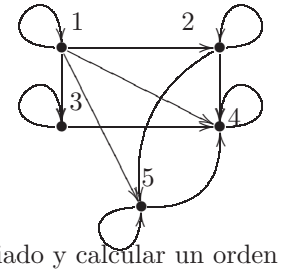
    Consideramos el subconjunto $S = \{c, d, e, f\}$. Dar una cota superior y una cota inferior de S. Calcular el $\sup(S)$ y el $\inf(S)$. Dar un orden total que contenga al orden parcial.

20. Dado el orden:



    Dibujar su diagrama de Hasse asociado y calcular un orden total que contenga al mismo.

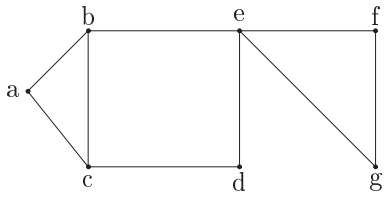
21. Dado el orden:



Dibujar su diagrama de Hasse asociado y calcular un orden total que contenga al mismo.

Grafos

22. Dado el grafo:



Calcular:

a) Un camino de b a d que no sea recorrido.

b) Un recorrido de b a d que no sea camino simple.

c) Un camino simple de b a d.

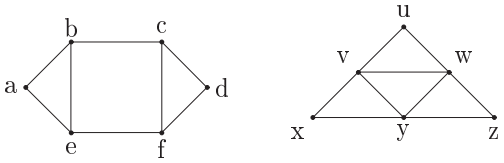
d) Un camino cerrado $b - b$ que no sea circuito.

e) Un circuito $b - b$ que no sea un ciclo.

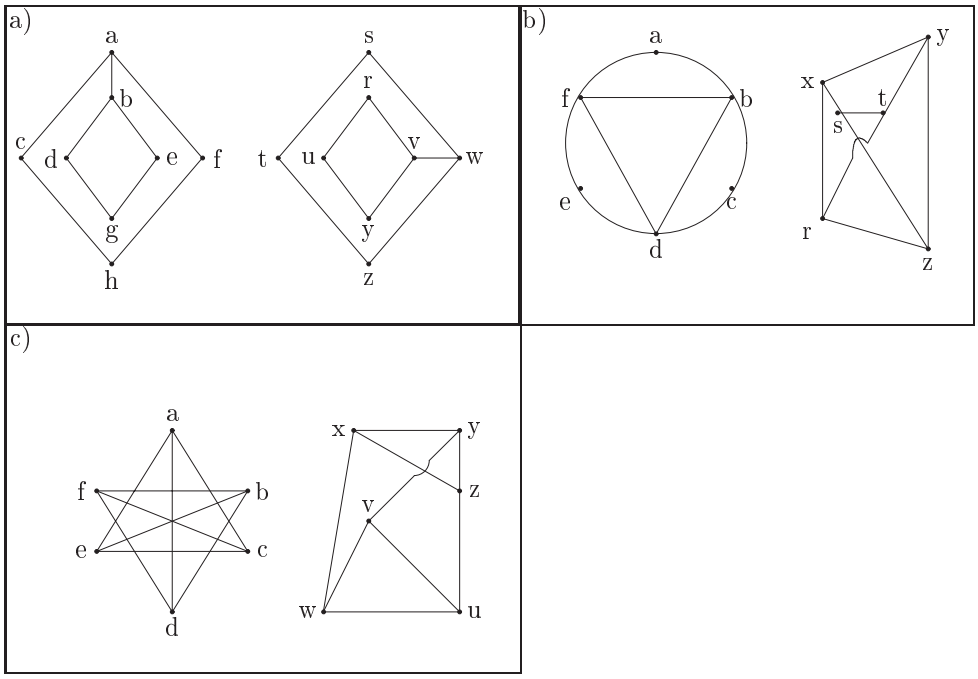
f) Un ciclo $b - b$.

g) Calcular el número de caminos simples que existen de b a f.

23. Demostrar que los siguientes grafos no son isomorfos:



24. Determinar si cada par de grafos son o no isomorfos:



25. Sea $V = \{000, 001, 010, \ldots, 110, 111\}$. Para cualquier sucesión de cuatro bits $b_1b_2b_3b_4$ trazar una arista del elemento $b_1b_2b_3$ al elemento $b_2b_3b_4$ en V.

    a) Trazar el grafo $G = (V, A)$ descrito.

    b) Encontrar un ciclo hamiltoniano dirigido para G.

    c) Distribuir ocho ceros y ocho unos de modo uniforme alrededor del borde de un disco que gira en el sentido de las manecillas del reloj de modo que estos 16 bits formen una sucesión circular tal que las subsucesiones (consecutivas) de longitud 4 proporcionen las representaciones binarias de $0, 1, 2, \ldots, 15$ en algún orden.

26. Carolina y Ricardo van a una fiesta con otras tres parejas. En esta fiesta hubo mucho apretones de manos pero, (1) nadie estrechó la mano de su pareja; (2) nadie estrechó su propia mano; y (3) nadie dio la mano más de una vez a otra persona. Antes de salir de la fiesta, Carolina preguntó a las otras 7 personas a cuántas habían dado la mano, y recibió una respuesta diferente de cada uno. ¿Cuántas veces dio Carolina la mano en esta fiesta? Cuántas veces lo hizo Ricardo?

27. Distribuir nueve ceros, nueve unos y nueve doses de modo uniforme alrededor del borde de un disco que gira en el sentido de las manecillas del reloj de modo que estos 27 símbolos formen una sucesión circular tal que las subsucesiones (consecutivas) de longitud 3 proporcionen las representaciones ternarias (base 3) de $0, 1, 2, \ldots, 25, 26$.

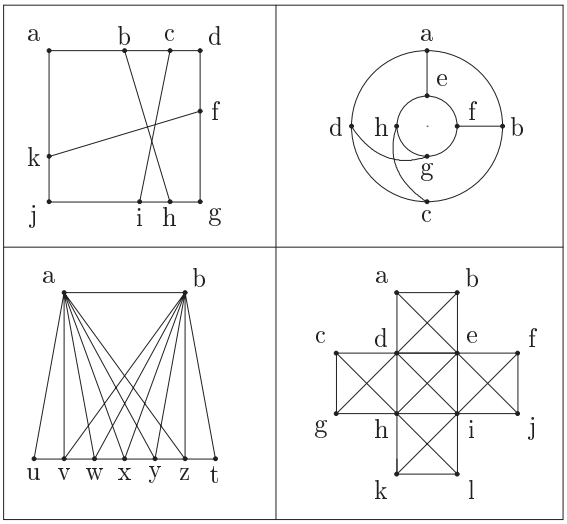
28. En los siguientes grafos se pide calcular:

    a) grados de los vértices

    b) $\kappa(G)$ =número de componentes conexas de G

    c) Un camino simple, un ciclo, un recorrido y un circuito.

    d) Decidir si tiene algún circuito o recorrido euleriano. Lo mismo para un camino o ciclo hamiltoniano.



29. *a*) Encontrar dos grafos $G = (V, E)$ y $G_1 = (V_1, E_1)$ con $v \in V$ y $v_1 \in V_1$ de modo que:

$$\kappa(G - v) = \kappa(G) \quad \text{pero} \quad \kappa(G_1 - v_1) > \kappa(G_1)$$

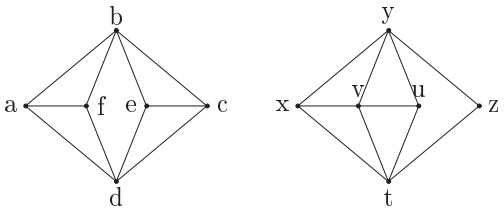
b) Encontrar dos grafos $G = (V, E)$ y $G_1 = (V_1, E_1)$ con $e \in E$ y $e_1 \in E_1$ de modo que:

$$\kappa(G - e) = \kappa(G) \quad \text{pero} \quad \kappa(G_1 - e_1) > \kappa(G_1)$$

30. Determinar los valores de n para los que el grafo completo K_n tiene un circuito euleriano. ¿ Para qué n tiene K_n un recorrido euleriano pero no un circuito euleriano?

31. Para cada $n \geq 3$, denotamos por C_n el ciclo de longitud n. Demostrar que $P(C_n, \lambda) = (\lambda - 1)^n + (-1)^n(\lambda - 1)$.

32. Dados los grafos:

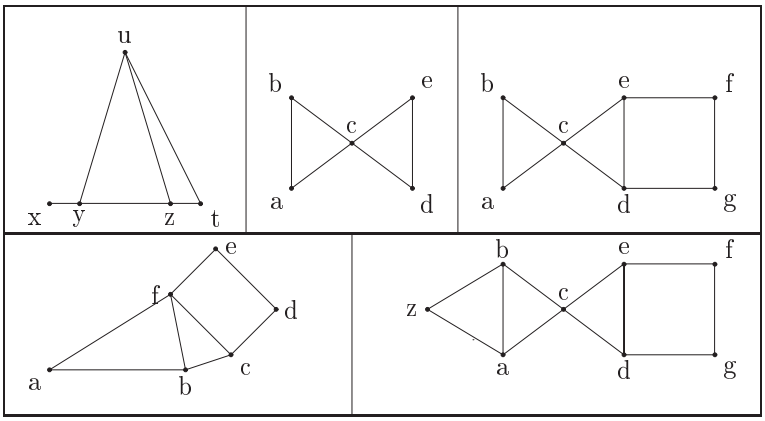


a) Determinar si son isomorfos.

b) Encontrar $P(G, \lambda)$ para cada grafo.

c) Comentar los resultados de los apartados anteriores.

33. Dados los grafos:



a) Determinar sus polinomios cromáticos.

b) Encontrar $\chi(G)$ para cada grafo.

c) Si se dispone de cinco colores, ¿ cuántas coloraciones propias de los vértices de cada grafo existen?

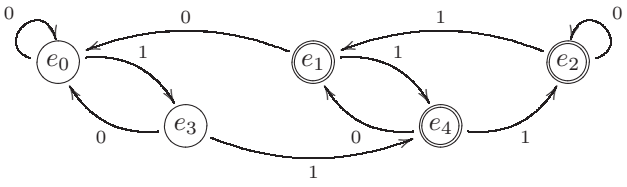
34. (Examen Febrero 2003) Para $n \geqslant 3$, sea $\mathcal{C}_n$ el ciclo de longitud n. Probar las siguientes relaciones:

$$P(\mathcal{C}_n, \lambda) - (\lambda - 1)^n = (\lambda - 1)^{n-1} - P(\mathcal{C}_{n-1}, \lambda),\ n \geqslant 4.$$
$$P(\mathcal{C}_n, \lambda) - (\lambda - 1)^n = P(\mathcal{C}_{n-2}, \lambda) - (\lambda - 1)^{n-2},\ n \geqslant 5.$$

35. En unos laboratorios químicos se reciben siete sustancias químicas $\{s_i\}_{i=1,\ldots,7}$ que precisan ser almacenadas. La naturaleza de las sustancias es tal que para $1 \leq i \leq 5$ la sustancia s_i no puede ser almacenada con la sustancias s_{i+1} y s_{i+2}. Determinar el número mínimo de compartimentos necesarios para almacenar las mencionadas sustancias químicas en las condiciones anteriores. Con ese número mínimo de compartimentos, ¿de cuántas maneras distintas se pueden almacenar las sustancias químicas?

Máquinas Finitas

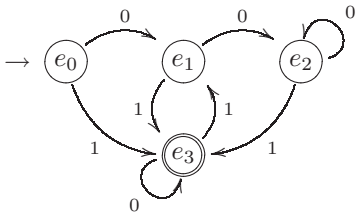
36. Dada la máquina finita:



a) Calcular $f_{e_0}(011011010)$.

b) Obtener todas las palabras de entradas α tales que $f_{e_0}(\alpha) = 0011110$.

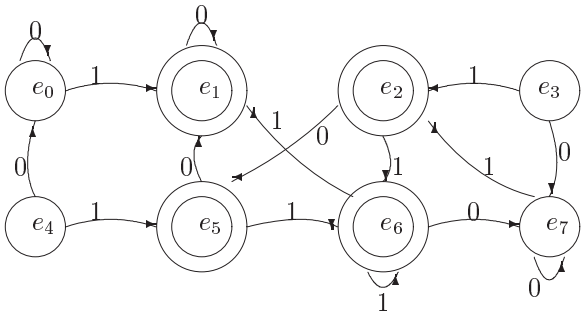
37. Dada la máquina siguiente, calcular $f_{e_0}(\alpha)$ para toda palabra de entradas α.



38. Construir una máquina finita lo más sencilla posible con $I = \{0, 1\}$ de manera que $f_{e_0}(\alpha) = 0\alpha$ para toda palabra de entrada α.

39. Demostrar que existe un único homomorfismo entre las siguientes máquinas que es epimorfismo pero no es monomorfismo.

40. Sea M la máquina cuyo grafo de estados es:

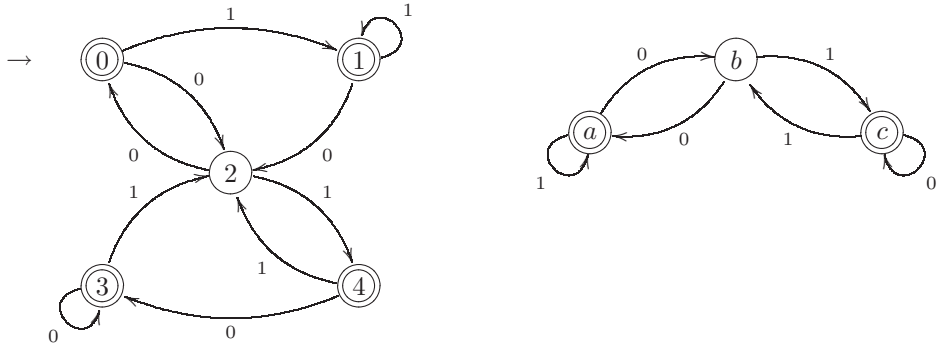


y R la relación en E cuya partición asociada es:

$$\{\{e_0, e_4\}, \{e_1, e_2, e_5\}, \{e_6\}, \{e_3, e_7\}\}$$

Dibujar el grafo de M/R.

41. Dadas las máquinas M y M', definir un homomorfismo $g\colon M \longrightarrow M'$, dibujar el grafo de M/g y definir el isomorfismo entre M/g e $\mathrm{Im}(g)$.



42. Determinar en cada caso si la palabra pertenece o no al conjunto regular representado por la expresión regular.

   a) 10100010 ; (0*10)*

   b) 011100 ; (0∨(11)*)*

   c) 000111100 ; ((001∨11)*(00)*)*

d) 01110111 ; (1*01)*(11∨0*)

e) 11100111 ; ((1*0)*∨0*11)*

f) 011100101 ; 01*10*(11*0)*

g) 1000011 ; (10*∨11)*(0*1)*

43. Determinar cuáles de las siguientes igualdades son ciertas.

a) $(a \vee b)^* = a^*b^*$

b) $(a \vee b)^* = (a^*b^*)^*$

c) $(a \vee b)^* = (a^* \vee b)^*$

d) $a^*a = aa^*$

e) $a^*b = aa^*b \vee b$

44. Dar una expresión regular para cada uno de los siguientes conjuntos regulares sobre $X = \{a, b\}$:

a) Palabras que empiezan por a.

b) Palabras que tienen exactamente una a.

c) Palabras que tienen un número par de aes.

d) Palabras que contienen la palabra aa.

45. Construir una máquina finita de Moore que reconozca el lenguaje L sobre $I = \{0, 1\}$ en cada caso.

a) Palabras que empiezan por cero.

b) Palabras que tienen exactamente un cero.

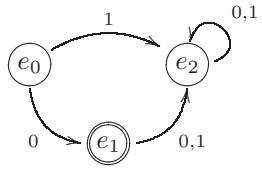
c) Palabras que tienen exactamente un cero y empiezan por él.

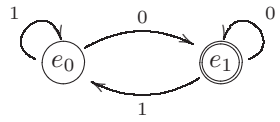
d) Palabras que terminan en 101.

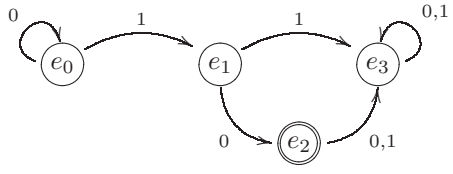
e) Palabras que terminan en 00.

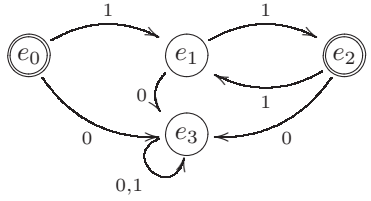
f) Palabras que contienen 10 ó 01.

46. Dar una expresión regular para $L(M)$ en los casos siguientes:

(a) 

(b) 

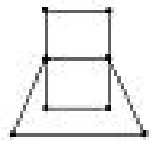
(c) 

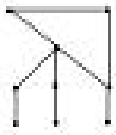
(d) 

Ejercicios de Grafos

2 curso I.T.I.S. Universidad de Salamanca 2010/11

1. Dibujar los grafos, la rueda W_3, el cubo Q_3, los grafos completos K_3, K_4 y los grafos bipartidos completos $K_{2,5}$, $K_{3,4}$.
2. Encontrar todos los grafos con cuatro vértices o menos no isomorfos entre sí.
3. Hallar la matriz de adyacencias de los grafos dibujados en el primer ejercicio.
4. Hallar todos los subgrafos de $K_{1,6}$ y de W_3.
5. Hallar los valores de n para los que los grafos K_n, C_n y W_n son grafos eulerianos.
6. En un mapa de carreteras aparecen 25 tramos de carretera. Sabiendo que en todos los cruces hay una población y que cada lugar tiene al menos cuatro caminos incidentes con él ¿Cuántas poblaciones aparecen en el mapa?
7. Se dispone de seis ordenadores y nueve cables de conexión. Se necesita conectar cada ordenador con otros tres. ¿Existe alguna forma de conectarlos? ¿Es única?
8. ¿Cuántas aristas tiene un grafo simple si sus vértices tienen los siguiente grados $4, 2, 2, 3, 3$? Dibuja ese grafo.
9. ¿Es posible encajar todas las fichas de un dominó en un tablero?
10. Determinar si los siguiente grafos son Hamiltonianos.





11. Dibujar los digrafos con lazos cuyas matrices de adyacencia son las siguiente:

$$\begin{pmatrix} 0 & 2 & 2 & 2 \\ 2 & 0 & 1 & 1 \\ 2 & 1 & 0 & 1 \\ 2 & 1 & 1 & 0 \end{pmatrix}, \quad \begin{pmatrix} 0 & 1 & 2 & 2 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 3 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{pmatrix}$$

12. Encontrar el camino mínimo entre A y L en el grafo etiquetado $G = (V, A, d)$, donde

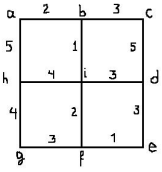
$$d : A \to \mathbb{R}$$

está dado por la siguiente tabla:

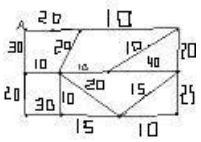
	A	B	C	D	E	F	G	H	I	J	K	L
A	-	3	2	-	8	-	-	-		-	-	-
B		-	-	2	4	-	-	-	-	-	-	-
C			-	-	6	9	-	-	-	-	-	-
D				-	-	-	2	-	-	-	-	-
E					-	-	1	2	-	-	-	-
F						-	-	1	2	-	-	-
G							-	-	-	6	-	-
H								-	-	5	6	9
I									-	-	2	-
J										-	-	5
K											-	3
L												-

Se considera que el grafo no es dirigido y que solo posee aristas etiquetados

13. Encontrar el camino más corto entre a y e en el siguiente grafo etiquetado.



14. El grafo de la figura representa un barrio de una ciudad donde la etiqueta de cada arista es la longitud de la calle correspondiente. El edificio A presenta problemas en su suministro de gas y estalla la conducción. La onda expansiva tiene un alcance de 65 metros y se propaga a través de las calles. ¿Cuáles son los edificios afectados?



15. A comienzo de un año una empresa debe comprar una nueva máquina. El costo de mantenimiento de este con i años de antigüedad, así como el costo de compra de una máquina al inicio de cada año se da en la siguiente tabla. El objetivo es minimizar el costo total (compra más mantenimiento) de tener una máquina durante cinco años. Determinar los años en los que se debe comprar una nueva máquina.

	mant.	compra
0	38 000	170 000
1	50 000	190 000
2	97 000	210 000
3	182 000	250 000
4	304 000	300 000

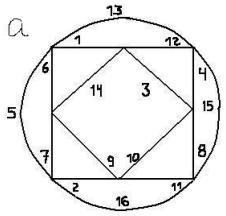
16. Utilizando el orden alfabético, construir un árbol de búsqueda binaria para las palabras Villamayor, Santa Marta, Salamanca, Cabrerizos, Monterrubio de Armuña, Villares de la Reina, Ledesma, Peñaranda de Bracamonte, Vitigudino.

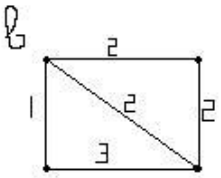
17. Sea

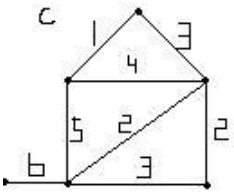
    $$\{\{1,2\},\{1,3\},\{1,4\},\{2,5\},\{2,6\},\{3,10\},\{3,11\},\{1,2\},\{11,12\},\{11,13\},\{5,7\},\{5,8\},\{8,9\},\}$$

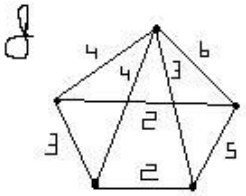
    la relación de equivalencia que define un grafo conexo en el conjunto de sus vértices $X_{13} = \{1,\dots 13\}$. Encontrar un árbol generador por el método de búsqueda de profundidad y de amplitud.

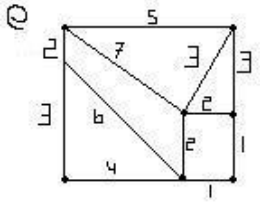
18. Encontrar un árbol generador de peso mínimo en los grafos siguientes:







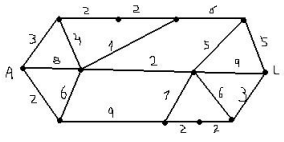




19. El grafo de la figura representa las posibilidades de instalación de cableado de una red eléctrica cuyo origen es la central situada en s_0.

    i. Describir un cableado que de abastecimiento a todos los barrios s_i de coste mínimo.

    ii. Supongamos ahora que en los barrios s_1, s_5 y s_9 se alojan personas muy influyentes que obligan a la empresa eléctrica a cablear toda la instalación que llega a esos barrios. Encontrar el coste mínimo para la empresa eléctrica para que todos los barrios dispongan de electricidad. ¿Cuánto es la cantidad que se derrocha teniendo en cuenta este apartado?

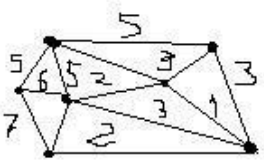


20. Se quiere renovar parte de los tramos de un ferrocarril que conectan 8 ciudades A, B, C, D, E, F, G y H. La duración estimada del viaje directo entre cada dos ciudades viene dado por la tabla siguiente

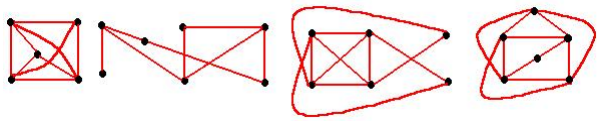
A	B	C	D	E	F	G	H
-	3	-	-	-	3	-	2
	-	5	-	-	-	3	-
		-	1	-	3	-	2
			-	4	-	1	-
				-	3	-	-
					-	2	-
						-	3
							-

Se requiere que dos estaciones cualesquiera queden conectadas por tramos renovados y que la duración del viaje de A a cualquier otra ciudad por tramos renovados sea lo menor posible. ¿Qué tramos, optimizando el trabajo y los coses, han de renovarse?

21. En el grafo de la figura se muestra una red de ordenadores que se quiere construir. Los vértices representan los ordenadores y las aristas las líneas de transmisión a considerar para conectar algunos pares de ellos. Cada arista tiene un peso que indica el coste de construir dicha línea. Conecta todos los ordenadores con el menor coste posible.

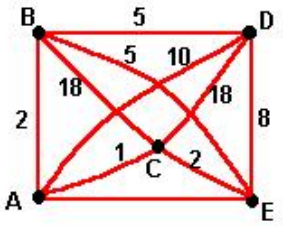


22. Calcular el número cromático para cada uno de los grafos siguiente:

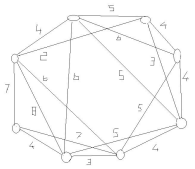


23. Ayuda, utilizando la búsqueda de profundidad y de anchura, a encontrar una solución al problema que tiene un vendedor, que debe recorrer las cinco ciudades del grafo, pasando exactamente un vez por cada una.

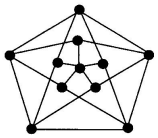
Si además necesita gastar lo mínimo, y el traslado desde una ciudad a otra es el del grafo, ¿Cuál sería una posible solución?



24. En la Facultad de Ciencias, Departamento de Informática se celebra todos los años un congreso de una semana sobre Teoría de Grafos. Este año se imparten ocho cursos de una hora diaria. Solo tienen tres horas disponibles cada día, pero no hay restricción en el número de aulas libres. En el grafo siguiente se tiene la información sobre el número de alumnos que comparten dos cursos. ¿Es posible planificar el horario de conferencias de manera que todos los alumnos puedan asistir a los cursos en que están matriculados? ¿Existe alguna forma de hacer el horario de manera que haya el mínimo número de alumnos perjudicados?



25. ¿El grafo siguiente es un grafo plano?



26. Encontrar los grafos completos planos.

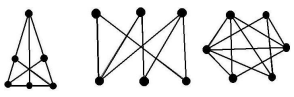
27. Demostrar que en cualquier grafo plano con $|V| \geq 3$ existe al menos un vértice con grado menor o igual a cinco.

28. Encontrar los números naturales n para los cuáles K_n es un grafo bipartido.

29. Demostrar que todo subgrafo de un grafo plano es plano.

30. Encontrar todos los subgrafos planos de $K_{3,3}$.

31. Determinar cuáles de los siguiente grafos son planos.



32. Determinar si K_5, $K_{3,3}$ y $H_{3,4}$ son planos si se les elimina una arista.

33. Sea $= (V, A)$ un grafo simple conexo con 16 aristas y tal que todo vértice es de grado al menos 4. Si G es plano, ¿cuántos vértices tiene? Dibuja un grafo con estas condiciones.

Ejercicios básicos de Códigos. Hoja 1.

2° Curso I.T.I.S. Universidad de Salamanca 2008/09

1. Para el código lineal C con función codificadora

$$c : \mathbb{Z}_2^3 \to \mathbb{Z}_2^5 \ (x, y, z) \mapsto c(x, y, z) := (x, x + y, z, x + y + z, y + z)$$

se pide:

   (a) Especificar todas las palabras de C usando la función c.

   (b) Encontrar una base de C y con ella una forma paramétrica de C.

   (c) Calcular una matriz generadora G y usarla para generar todas las palabras de C.

   (d) Calcular una forma implícita de C y una matriz de control H.

   (e) Calcular la distancia d(C) del código.

   (f) Especificar la longitud n, la dimensión k y la distancia d.

   (g) Descodificar por distancia mínima la palabra $y_0 = (11111)$.

   (h) Descodificar por síndromes la palabra y_0 anterior.

2. Sea C el $(6, 3)$-código lineal sobre $\mathbb{Z}_2$ con matriz generadora $G = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$

   (a) Especificar todas las palabras de C.

   (b) Calcular la distancia mínima de C.

   (c) Recibidas las palabras 011000 y 101100, detectar si se ha producido un error en la transmisión y, en su caso, aplicar el método de decodificación por distancia mínima.

3. Considérese el $(5, 2)$-código binario C con matriz generadora $G = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{pmatrix}$.

   (a) Calcular el peso de C.

   (b) Dado el vector $u = 10001 \in \mathbb{Z}_2^5$, ¿cuál es el líder de su clase?

4. Para el código lineal C con función codificadora

$$c : \mathbb{Z}_2^3 \to \mathbb{Z}_2^6 \ (x, y, z) \mapsto c(x, y, z) := (x, y, x + y + z, x + y, y + z, x + z)$$

se pide:

   (a) Especificar todas las palabras de C usando la función c.

   (b) Encontrar una base de C y con ella una forma paramétrica de C.

   (c) Calcular una matriz generadora G y usarla para generar todas las palabras de C.

   (d) Calcular una forma implícita de C y una matriz de control H.

(e) Calcular la distancia $d(C)$ del código.

(f) Especificar la longitud n, la dimensión k y la distancia d.

(g) Descodificar por distancia mínima la palabra $y_0 = (101010)$.

(h) Descodificar por síndromes la palabra y_0 anterior.

5. Sea C el $(6,3)$-código lineal sobre $\mathbb{Z}_2$ con matriz generadora

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

(a) Especificar todas las palabras de C. Calcular la distancia mínima de C.

(b) Recibidas las palabras 010100 y 110100, detectar si se ha producido un error en la transmisión y, en su caso, aplicar el método de decodificación por distancia mínima.

6. Consideremos el $(5,3)$-código binario C con matriz generadora y, en su caso, aplicar el método de decodificación por distancia mínima.

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

(a) Calcular el peso de C. Dado el vector $u = 10101 \in \mathbb{Z}_2^5$, ¿cuál es el líder de su clase?

7. Sea el código bloque $(7,4)$ generado por la matriz $G = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

(a) Encontrar todas las palabras del código C.

(b) ¿Cuáles son las capacidades de corrección y detección de C?

(c) Encontrar la matriz de control H de C.

(d) Construir la tabla de síndromes de C.

(e) Comprobar el síndrome de la palabra recibida 1101101. ¿Es una palabra del código? Si la respuesta es no, ¿Cuál es la palabra que mayor probabilidad tiene de haber sido la enviada?

8. Sea $\begin{pmatrix} 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix}$ la matriz H de paridad de un código C.

(a) Construir la tabla de decodificación por mínima distancia con síndromes.

(b) Usar la tabla para decodificar las siguientes palabras

11110 11101 11011 10100 10011 10101 11111 01100

9. ¿Cuál es la dimensión de la matriz generadora del código de Hamming $(63, 57)$? ¿Y de la matriz de paridad?

10. ¿Cuál es la razón de codificación de este código? ¿Y la tasa de codificación?