5 GRAFOS

5.1 GRAFOS Y SUS REPRESENTACIONES

Los objetos a los cuales llamaremos *grafos* son muy útiles en matemática discreta. Su nombre se deriva del hecho de que pueden ser entendidos con una notación gráfica (o pictórica), y en este aspecto solamente se parecen a los familiares gráficos de funciones que son estudiados en matemática elemental. Pero nuestros grafos son bastante diferentes de los gráficos de funciones y están más relacionados con objetos que en el lenguaje diario llamamos *redes* (networks).

Usaremos la siguiente definición en lo que sigue: dado un conjunto X un *2-subconjunto* es un subconjunto de X de dos elementos.

Definición 5.1.1. Un *grafo* G consiste de un conjunto finito V, cuyos miembros son llamados *vértices*, y un conjunto de 2-subconjuntos de V, cuyos miembros son llamados *aristas*. Nosotros usualmente escribiremos $G = (V, E)$ y diremos que V es el *conjunto de vértices* y E es el *conjunto de aristas*.

La restricción a un conjunto finito no es esencial, pero es conveniente para nosotros debido a que no consideraremos "grafos infinitos" en este apunte.

Un ejemplo típico de un grafo $G = (V, E)$ es dado por los conjuntos

$$V = \{a, b, c, d, z\}, \qquad E = \{\{a, b\}, \{a, d\}, \{b, z\}, \{c, d\}, \{d, z\}\}. \tag{5.1.1}$$

Este ejemplo y la definición misma no son demasiado esclarecedores, y solamente cuando consideramos la *representación pictórica* de un grafo es cuando se hace la luz.

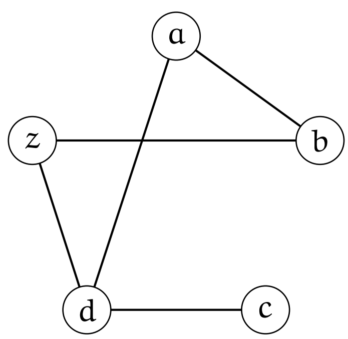

Figura 4: Una representación pictórica del grafo definido en (5.1.1).

Nosotros representamos los vértices como puntos, y unimos dos puntos con una linea siempre y cuando el correspondiente par de vértices está en una arista. Luego la Fig. 4 es una representación pictórica del grafo dado en el ejemplo arriba. Esta clase de representación es extremadamente conveniente para trabajar "a mano" con grafos relativamente pequeños. Más aún, esta representación es de gran ayuda para formular y comprender argumentos abstractos. Nosotros damos a continuación un ejemplo frívolo.

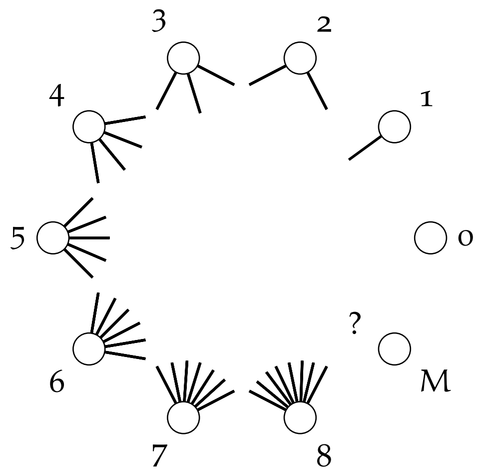

Figura 5: La fiesta de Abril

Ejemplo. Mario y su mujer Abril dan una fiesta en la cual hay otras cuatro parejas de casados. Las parejas, cuando arriban, estrechan la mano a algunas personas, pero, naturalmente, no se estrechan la mano entre marido y mujer. Cuando la fiesta finaliza el profesor pregunta a los otros a cuantas personas han estrechado la mano, recibiendo 9 respuestas diferentes. ¿Cuántas personas estrecharon la mano de Abril?

Solución. Construyamos un grafo cuyos vértices son las personas que asisten a la fiesta. Las aristas del grafo son las $\{x, y\}$ siempre y cuando x e y se hayan estrechado las manos. Puesto que hay nueve personas aparte de Mario, y que una persona puede estrechar a lo sumo a otras 8 personas, se sigue que las 9 respuestas diferentes que ha recibido el profesor deben ser $0, 1, 2, 3, 4, 5, 6, 7, 8$. Denotemos los vértices con estos números y usemos M para Mario. Así obtenemos la representación pictórica de la Fig. 5

Ahora, el vértice 8 alcanza a todos los otros vértices excepto uno, el cual debe por lo tanto representar a la esposa de 8. Este vértice debe ser el 0 el cual por cierto que no está unido al 8 (ni obviamente a ningún otro). Luego 8 y 0 son una pareja de casados y 8 está unido a $1, 2, 3, 4, 5, 6, 7$ y M. En particular el 1 está unido al 8 y ésta es la única arista que parte del 1. Por consiguiente 7 no esta unido al 0 y al 1 (únicamente), y la esposa de 7 debe ser 1, puesto que 0 esta casado con 8. Continuando con este razonamiento vemos que 6 y 2, y 5 y 3 son parejas de casados. Se sigue entonces que M y 4 están casados, luego el vértice 4 representa a Abril, quien estrechó la mano de cuatro personas. □

Aunque la representación pictórica es intuitivamente atractiva para los seres humanos, es claramente inútil cuando deseamos comunicarnos con una computadora. Para lograr esto debemos representar el grafo mediante cierta clase de lista o tabla. Diremos que dos vértices x e y de un grafo son *adyacentes* cuando $\{x, y\}$ es una arista. (o también diremos que x e y son *vecinos*). Entonces podemos representar un grafo $G = (V, E)$ por su *lista de adyacencia*, donde cada vértice v encabeza una lista de aquellos vértices que son adyacentes a v. El grafo de Fig. 4 tiene la siguiente lista de adyacencia:

a	b	c	d	z
b	a	d	a	b
d	z		c	d
			z	

Las listas de adyacencia son redundantes (cada arista está representada dos veces) pero como todo lenguaje de programación de alto nivel maneja la estructura tipo lista, preferimos esta representación pues un grafo resulta ser una lista de listas o un arreglo de listas.

Ejemplo. Por cada entero positivo n definimos el *grafo completo* K_n como el grafo con n vértices y en el cual cada par de vértices es adyacente.

¿Cuántas aristas tiene K_n? De cada vértice "salen" $n-1$ aristas, las que van a otros vértices. Si sumamos n-veces las $n-1$ aristas es claro que estamos contando cada arista dos veces, luego el número total de aristas es $n(n-1)/2$ (observar que esta es una demostración, usando grafos, de que $\sum_{i=1}^{n} i = n(n-1)/2$).

§ *Ejercicios*

1) A tres casas A, B, C se les debe conectar el gas, el agua y la electricidad: G, W, E. Escribir la lista de adyacencia para el grafo que representa este problema y construir una representación pictórica del mismo. ¿Puede usted encontrar un dibujo en el cual las líneas que representan las aristas no se crucen?

2) Los senderos de un jardín han sido diseñados dándoles forma de *grafo rueda* W_n, cuyos vértices son $V = \{0, 1, 2, \ldots, n\}$ y sus aristas son

$$\begin{array}{lll} \{0,1\}, & \{0,2\}, \ldots, \{0,n\}, & \\ \{1,2\}. & \{2,3\}, \ldots, \{n-1,n\}, & \{n,1\}. \end{array}$$

Describir una ruta por los senderos de tal forma que empiece y termine en le vértice 0 y que pase por cada vértice una sola vez.

3) ¿Para cuales valores de n se puede hacer una representación pictórica de K_n con la propiedad que las líneas que representan las aristas no se corten?

4) Un *3-ciclo* en un grafo es un conjunto de tres vértices mutuamente adyacentes. Construir un grafo con cinco vértices y seis aristas que no contenga 3-ciclos.

5.2 ISOMORFISMO DE GRAFOS

En este punto nosotros debemos enfatizar que un grafo esta definido como una entidad matemática abstracta. Es en este contexto que nosotros discutiremos el importante problema de que queremos decir cuando decimos que dos grafos son "el mismo".

Claramente lo importante de un grafo no son los nombres con que designamos a los vértices, ni su representación pictórica o cualquier otra representación. La propiedad característica de un grafo es la manera en que los vértices están conectados por aristas.

Antes de definir isomorfismo de grafos repasaremos el concepto de función o aplicación biyectiva. Dado dos conjuntos X, Y diremos que una aplicación $f : X \to Y$ es *biyectiva* si para cada $y \in Y$ existe un único $x \in X$ tal que $f(x) = y$. Un propiedad importante, de las funciones biyectivas es que f es biyectiva si y sólo sí f tiene *inversa*, es decir existe $f^{-1} : Y \to X$, tal que $f(f^{-1}(y)) = y$, $\forall y \in Y$ y $f^{-1}(f(x)) = x$, $\forall x \in X$.

Ejemplo. La función

$$f : \{1, 2, 3\} \to \{a, b, c\} \quad \text{definida } f(1) = c, f(2) = b, f(3) = a$$

es biyectiva y su inversa es

$$f^{-1}(a) = 3,\ f^{-1}(b) = 2,\ f^{-1}(c) = 1.$$

También es biyectiva la aplicación

$$g : \{x, y\} \times \{u, w, z\} \to \{1, 2, 3, 4, 5, 6\} \quad \text{definida}$$

$$g(x, u) = 1,\ g(x, w) = 2,\ g(x, z) = 3,\ g(y, u) = 4,\ g(y, w) = 5,\ g(y, z) = 6.$$

Definición 5.2.1. Dos grafos G_1 y G_2 se dicen que son *isomorfos* cuando existe una biyección α entre el conjunto de vértices de G_1 y el conjunto de vértices de G_2 tal que si $\{x, y\}$ es una arista de G_1 entonces $\{\alpha(x), \alpha(y)\}$ es una arista de G_2 y recíprocamente si $\{z, w\}$ es una arista de G_2 entonces $\{\alpha^{-1}(z), \alpha^{-1}(w)\}$ es una arista de G_1. La biyección α es llamada un *isomorfismo*.

Por ejemplo, considere los dos grafos de la Fig. 6. En este caso hay una biyección entre el conjunto de vértices de G_1 y el conjunto de vértices de G_2 la cual tiene la propiedad requerida; esta biyección es dada por

$$\alpha(a) = t, \quad \alpha(b) = v, \quad \alpha(c) = w, \quad \alpha(d) = u.$$

Podemos comprobar que a cada arista de G_1 le corresponde una arista de G_2 y viceversa. Por ejemplo, a la arista bc de G_1 le corresponde la arista vw de G_2, y así siguiendo. (Usaremos la abreviación xy para la arista $\{x, y\}$, recordando que una arista es un par desordenado, es decir xy es lo mismo que yx.)

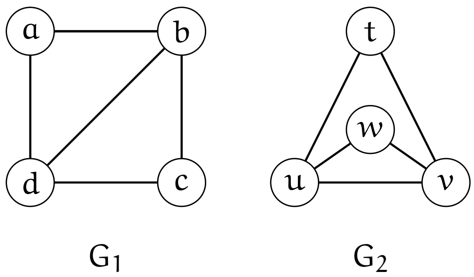

Figura 6: G_1 y G_2 son isomorfos

Cuando, como en la Fig. 6, dos grafos G_1 y G_2 son isomorfos usualmente nos referiremos a ellos como que son "el mismo" grafo.

Para mostrar que dos grafos no son isomorfos, nosotros debemos demostrar que no hay una biyección entre el conjunto de vértices de uno con el conjunto de vértices de otro, que lleve las aristas de uno en las aristas del otro.

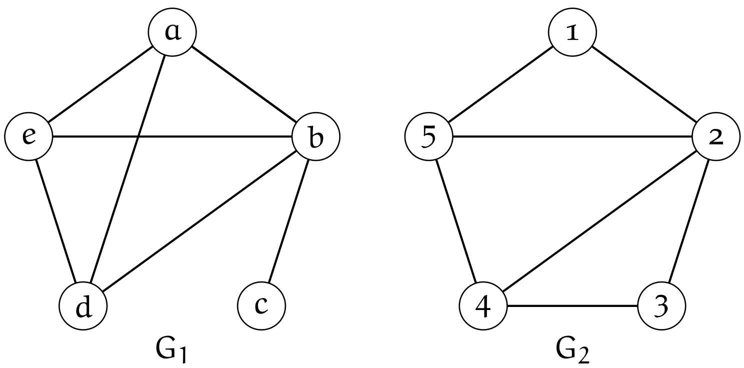

Figura 7: G_1 y G_2 no son isomorfos

Si dos grafos tienen diferente número de vértices, entonces no es posible ninguna biyección, y los grafos no pueden ser isomorfos. Si los grafos tienen el mismo número de vértices, pero diferente número de aristas, entonces hay biyecciones de vértices pero ninguna de ellas puede ser un isomorfismo.

Definición 5.2.2. Sea $G = (V, E)$ un grafo. Se dice que $G' = (V', E')$ es *subgrafo* de $G = (V, E)$ si $V' \subset V$, $E' \subset E$ y todos los vértices que son extremos de las aristas de E' están en V'.

Es claro, pero no lo demostraremos aquí, que un isomorfismo lleva un subgrafo a un subgrafo isomorfo. Este resultado es una herramienta que puede ser útil para ver si dos grafos no son isomorfos.

Por ejemplo, los dos grafos de la Fig. 7 tienen cada uno cinco vértices y siete aristas pero no son isomorfos. Una manera de ver esto es observar que los vértices a, b, d, e forman un subgrafo completo de G_1 (cada par de ellos está conectado por una arista). Cualquier isomorfismo debe llevar estos vértices en cuatro vértices de G_2 con la misma propiedad, y puesto que no hay tal conjunto de vértices en G_2 no puede haber ningún isomorfismo.

§ *Ejercicios*

1) Probar que los grafos mostrados en la Fig. 8 no son isomorfos.

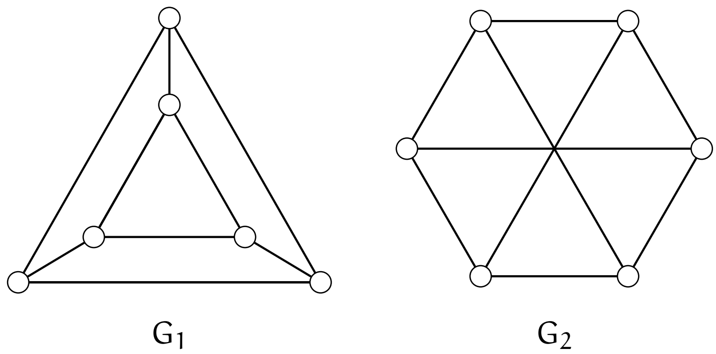

Figura 8: Probar que estos grafos no son isomorfos

2) Encontrar un isomorfismo entre los grafos definidos por las siguientes listas de adyacencias. (Ambas listas especifican versiones de un grafo famoso conocido como *grafo de Petersen*).

a	b	c	d	e	f	g	h	i	j
b	a	b	c	d	a	b	c	d	e
e	c	d	e	a	h	i	j	f	g
f	g	h	i	j	i	j	f	g	h

0	1	2	3	4	5	6	7	8	9
1	2	3	4	5	0	1	0	2	6
5	0	1	2	3	4	4	3	5	7
7	6	8	7	6	8	9	9	9	8

3) Sea $G = (V, E)$ el grafo definido como sigue: el conjunto de vértices V es el conjunto de todas las palabras de longitud tres en el alfabeto $\{0, 1\}$, y el conjunto de aristas E contiene aquellos pares de palabras que difieren exactamente en una posición. Probar que G es isomorfo al grafo formado por las esquinas y aristas de un cubo.

5.3 VALENCIAS DE UN GRAFO

La *valencia* o *grado* de un vértice v en un grafo $G = (V, E)$ es el número de aristas de G que contienen a v. Usaremos la notación $\delta(v)$ para la valencia de v, formalmente

$$\delta(v) = |D_v|, \quad \text{donde} \quad D_v = \{e \in E | v \in e\}.$$

El grafo descrito en Fig. 4 tiene $\delta(a) = 2$, $\delta(b) = 2$, $\delta(c) = 1$, $\delta(d) = 3$, $\delta(z) = 2$. El primer teorema de la teoría de grafos nos dice que la suma de estos números es dos veces el número de aristas.

Teorema 5.3.1. *La suma de los valores de las valencias $\delta(v)$, tomados sobre todos los vértices v del grafo $G = (V, E)$, es igual a dos veces el número de aristas:*

$$\sum_{v \in V} \delta(v) = 2|E|.$$

Demostración. La valencia de un vértice v indica la cantidad de "extremos" de aristas que "tocan" a v. Es claro que hay $2|E|$ extremos de aristas, luego la suma total de las valencias de los vértices es $2|E|$. □

Hay un útil corolario de este resultado. Diremos que un vértice de G es *impar* si su valencia es impar, y *par* si su valencia es par. Denotemos V_i y V_p los conjuntos de vértices impares y pares respectivamente, luego $V = V_i \cup V_p$ es una partición de V. Por teorema 5.3.1, tenemos que

$$\sum_{v \in V_i} \delta(v) + \sum_{v \in V_p} \delta(v) = 2|E|.$$

Ahora cada término en la segunda suma es par, luego esta suma es un número par. Puesto que el lado derecho también es un número par, la primera suma debe ser también par. Pero la suma de números impares solo puede ser par si el número de términos es par. En otra palabras:

Teorema 5.3.2. *El número de vértices impares es par.*

Este resultado es a veces llamado el "handshaking lemma" (handshak = estrechar la mano, darse la mano), debido a que se puede interpretar en términos de gente y darse la mano: dado un conjunto de personas, el número de personas que le ha dado la mano a un número impar de miembros del conjunto es par.

Un grafo en el cual todos los vértices tienen la misma valencia r se llama *regular* (con valencia r), o r-*valente,* o de *grado* r. En este caso, el resultado del teorema 5.3.1 se traduce a

$$r|V| = 2|E|.$$

Muchos de los grafos que aparecen en las aplicaciones son regulares. Ya conocemos los grafos completos K_n; ellos son regulares, con valencia $n-1$. De geometría elemental conocemos los polígonos de n lados, los cuales en teoría de grafos son llamados *grafos cíclicos* C_n. Formalmente, podemos decir que el conjunto de vértices de C_n es $\mathbb{Z}_n$, y los vértices i y j están unidos si $j = i + 1$ o $j = i - 1$ en $\mathbb{Z}_n$. Claramente, C_n es un grafo regular con valencia 2, si $n \geqslant 3$.

Una aplicación importante de la noción de valencia es en el problema de determinar si dos grafos son o no isomorfos. Si $\alpha : V_1 \to V_2$ es un isomorfismo entre G_1 y G_2, y $\alpha(v) = w$, entonces cada arista que contiene a v se transforma en una arista que contiene a w. En consecuencia $\delta(v) = \delta(w)$. Por otro lado, si G_1 tiene un vértice x, con valencia $\delta(x) = \delta_0$, y G_2 no tiene vértices con valencia δ_0, entonces G_1 y G_2 no pueden ser isomorfos. Esto nos da otra manera para distinguir los grafos de la Fig 7, puesto que el primer grafo tiene un vértice de valencia 1 y el segundo no.

Una extensión de esta idea se da en la siguiente proposición.

Proposición 5.3.3. *Sean* G_1 *y* G_2 *grafos isomorfos. Para cada* $k \geqslant 0$ *sea* $n_i(k)$ *el número de vértices de* G_i *que tienen valencia* k *(*$i = 1, 2$*). Entonces* $n_1(k) = n_2(k)$.

Demostración. Hemos visto más arriba que si $\alpha : V_1 \to V_2$ es un isomorfismo entre G_1 y G_2 y $v \in V_1$, entonces $\delta(v) = \delta(\alpha(v))$. Luego la cantidad de vértices con valencia k en G_1 es igual a la cantidad de vértices con valencia k en G_2. $\square$

Ejemplo. Revisemos los grafos de la Fig. 7 y la Fig. 8 de la sección anterior.

Los dos grafos de la Fig. 7 no son isomorfos debido a que en el primer grafo existen tres vértices con valencia 3 mientras que en el segundo existen sólo dos.

Observar que los criterios vistos hasta ahora relativos a cantidad de vértices, cantidad de aristas y valencias, incluyendo el de la proposición 5.3.3, no son útiles para determinar si los grafos de la Fig. 8 son isomorfos o no: ambos tienen 6 vértices, 9 aristas y todos los vértices son de valencia 3. Sin embargo, en el caso de la Fig. 8 podemos determinar que los grafos no son isomorfos observando los subgrafos de cada uno. Ahora bien, no hay ningún criterio general eficiente para determinar si dos grafos son isomorfos o no: en los casos difíciles esencialmente debemos probar con todas las biyecciones posibles de los vértices de un grafo a los vértices del otro y eso es no computable para casos no demasiado grandes.

§ *Ejercicios*

1) ¿Es posible que las siguientes listas sean las valencias de todos los vértices de un grafo? Si así lo fuera, dar una representación pictórica

de tal grafo. (Recordar que hay a lo más una arista que una un par de vértices dados.)

a) $2, 2, 2, 3$. *b)* $1, 2, 2, 3, 4$.

c) $2, 2, 4, 4, 4$. *d)* $1, 2, 3, 4$.

2) Si $G = (V, E)$ es un grafo, el *complemento* G^c de G es el grafo cuyo conjunto de vértices es V y cuyas aristas unen aquellos vértices que no son unidos por G. Si G tiene n vértices y sus valencias son $d_1, d_2, \ldots, d_n$, ¿cuáles son las valencias de G^c?

3) Encuentrar todos los grafos posibles (no isomorfos) que pueda, que sean regulares, 4-valentes y con 7 vértices. [Ayuda: considere el complemento de esos grafos.]

4) Probar que si G es un grafo con al menos dos vértices, entonces G tiene dos vértices con la misma valencia.

5.4 CAMINOS Y CICLOS

Frecuentemente usamos grafos como modelos de situaciones prácticas que involucran rutas: los vértices representan ciudades o cruces, y cada arista representa una ruta o cualquier otro forma de comunicación. Las definiciones de esta sección se comprenderán mejor con esta clase de ejemplo en mente.

Definición 5.4.1. Una *caminata* en un grafo G es una secuencia de vértices

$$v_1, v_2, \ldots, v_k,$$

tal que v_i y v_{i+1} son adyacentes ($1 \leqslant i \leqslant k-1$). Si todos los vértices son distintos, una caminata es llamada un *camino*.

Llamaremos *ciclo* a una caminata $v_1, v_2, \ldots, v_{r+1}$ con $r \geqslant 3$ y cuyos vértices son distintos exceptuando los extremos, es decir que $v_1, v_2, \ldots, v_r$ es un camino de al menos tres vértices y $v_1 = v_{r+1}$. A menudo diremos que es un r-*ciclo*, o un ciclo de *longitud* r en G.

Es decir, una caminata especifica una ruta en G: del primer vértice vamos a uno adyacente, de éste a otro adyacente y así siguiendo. En una caminata podemos visitar cualquier vértice varias veces, y en particular, podemos ir de un vértice x a otro y y luego tomar la dirección contraria y regresar a x. En un camino, cada vértice es visitado solo una vez.

Escribamos $x \sim y$ siempre y cuando los vértices x e y de G puedan ser unidos por un camino en G: hablando en forma rigurosa, esto significa que hay un camino $v_1, v_2, \ldots, v_k$ en G con $x = v_1$ e $y = v_k$.

Lema 5.4.2. *Sea* G *un grafo.* $x \sim y$ *si y sólo si* x *e* y *pueden ser unidos por una caminata*

Demostración. Es claro que si x e y están unidos por un camino, como un camino es un caso especial de caminata, x e y están unidos por una caminata.

Veamos que si x e y están unidos por una caminata, entonces están unidos por un camino. Sea

$$x = x_1, x_2, \ldots, x_k = y,$$

una caminata entre x e y. Si ninguno de los x_i se repite, entonces tenemos un camino y terminado el problema. Si hay repetición, entonces existe j tal que $x_j = x_{j+r}$ con $r > 0$, es decir tenemos una caminata

$$x = x_1, x_2, \ldots, x_j, \ldots, x_{j+r}, \ldots, x_k = y,$$

Como $x_j = x_{j+r}$ podemos eliminar la subcaminata $x_{j+1}, \ldots, x_{j+r}$ (un "bucle" dentro de la caminata) y nos queda

$$x = x_1, x_2, \ldots, x_j, x_{j+r+1}, \ldots, x_k = y,$$

una caminata, más corta, entre x e y. Podemos repetir este procedimiento hasta eliminar todos los "bucles" y obtener un camino. □

Definición 5.4.3. Diremos que un grafo G es *conexo* si para cualesquiera dos vértices x, y existe una camina de x a y, es decir si $x \sim y$.

Debido al lema que probamos más arriba, es sencillo verificar la siguientes propiedades: sea G grafo y sean x, y, z vértices de G, entonces

a) $x \sim x$ (reflexividad de $\sim$).

b) $x \sim y$, entonces $y \sim x$ (simetría de $\sim$).

c) $x \sim y$, $y \sim z$, entonces $x \sim z$ (transitividad de $\sim$).

En un lenguaje formal, una relación que cumple las tres propiedades anteriores es llamada una *relación de equivalencia* del conjunto, en este caso tenemos una relación de equivalencia del conjunto de vértices V de G. Como ya vimos en el capítulo 4, página 66, la congruencia módulo es también una relación de equivalencia, y hay muchísimos ejemplos en matemática de estos tipos de relaciones.

Estas tres propiedades que posee la relación de equivalencia permiten partir a V en conjuntos disjuntos: dos vértices están en el mismo conjunto si ellos pueden ser unidos por un camino, y están en conjuntos diferentes si no podemos encontrar tal camino. llamaremos a estos conjuntos disjuntos las *clases de equivalencia de* $\sim$.

Definición 5.4.4. Supongamos que $G = (V, E)$ es un grafo y que la partición de V en las clases de equivalencia de $\sim$ es

$$V = V_1 \cup V_2 \cup \cdots \cup V_r.$$

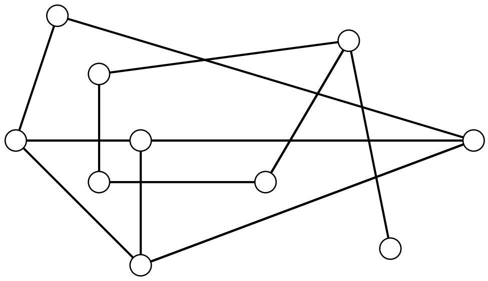

Figura 9: Un grafo con dos componentes

Denotemos con E_i $(1 \leqslant i \leqslant r)$ al subconjunto de E que contiene todas las aristas cuyos finales están en V_i. Entonces los grafos $G_i = (V_i, E_i)$ son llamados las *componentes* de G. Si G tiene solo una componente entonces, claramente, el grafo es conexo.

La terminología casi explica por si misma el significado de estas definiciones. El grafo mostrado en la Fig. 9 tiene dos componentes, y por consiguiente no es conexo. La descomposición de un grafo en componentes es muy útil, puesto que muchas propiedades de los grafos pueden ser establecidas considerando las componentes separadamente. Por esta razón, teoremas acerca de grafos a menudo son probados solo para la clase de grafos conexos.

Cuando un grafo de moderado tamaño es dado por una representación pictórica es bastante fácil determinar si es o no conexo. Sin embargo, cuando un grafo es dado por una lista de adyacencia necesitaremos un algoritmo eficiente para decidir si es o no conexo.

Ejemplo 5.4.5. Leandro y Juan, dos amigos, planean tomar sus vacaciones en determinada isla. La Fig. 10 representa los lugares de interés turístico de la isla y las carreteras que los unen. Leandro es un turista por naturaleza, y desea visitar cada lugar una vez y volver al punto de partida. Juan es un explorador, y desea atravesar todos los caminos solo una vez, a él lo tiene

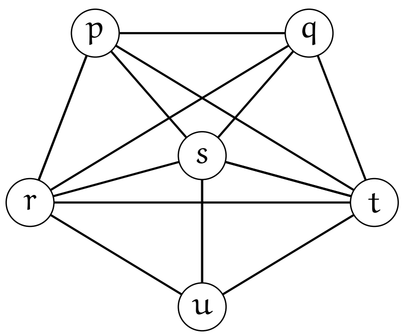

Figura 10: El gran tour

sin cuidado si regresa o no al lugar del cual partió. ¿Podrán encontrar las rutas que desean Leandro y Juan?

Solución. Leandro puede usar diferentes rutas para alcanzar su objetivo: una posibilidad es el ciclo p, q, t, s, u, r, p.

Sin embargo, Juan está en un apuro. Llamemos x al punto de partida y llamemos y al punto de llegada, y supongamos por el momento que $x \neq y$. Entonces él usa una arista con extremo en x para partir y cada vez que vuelve a x debe arribar y partir por nuevas aristas. Luego, usa un número impar de aristas con extremo en x, y por consiguiente x debe ser un vértice impar. De manera análoga, y debe ser también un vértice impar, puesto que Juan usa dos aristas cada vez que pasa por y, y una más al finalizar en y. Los restantes vértices deben ser pares, puesto que cada vez que Juan llega a un vértice intermedio parte de nuevo, y por consiguiente usa dos aristas.

Resumiendo, una ruta para Juan que empiece y finalice en vértices distintos x e y, es solo posible si hay dos vértices impares (que son x e y) y el resto de los vértices es par. Pero en el grafo de la Fig. 10 el valor de las valencias es: $\delta(p) = 4$, $\delta(q) = 4$, $\delta(r) = 5$, $\delta(s) = 5$, $\delta(t) = 5$, y $\delta(u) = 3$. Luego hay demasiados vértices impares, y por lo tanto no existe la ruta que Juan desea. Si permitimos la posibilidad de que $x = y$, la situación es aún peor, pues en este caso todos los vértices deberían ser pares. $\square$

En general, la ruta de Leandro es un ciclo que contiene todos los vértices del grafo dado. Tales ciclos fueron estudiados por el matemático irlandés W.R. Hamilton (1 805 – 65), y en consecuencia un ciclo con esta propiedad es llamado un *ciclo hamiltoniano*. En nuestro ejemplo, fue muy fácil encontrar un ciclo hamiltoniano, pero este fue un caso muy especial y no representativo. Para ciertos grafos, puede ser un problema difícil decidir si un ciclo hamiltoniano existe o no.

Por otro lado, el problema de Juan puede ser fácilmente resuelto. Una caminata que use cada arista de un grafo solo una vez es llamada una *caminata euleriana*, debido a que Euler fue el primero en estudiar estas caminatas y encontró que si $x \neq y$, una condición necesaria para que exista una caminata euleriana que comience en x y finalice en y es que x e y deben ser vértices impares y el resto debe ser par, mientras que si $x = y$ la condición es que todos los vértices deben ser pares. Es decir que una condición necesaria para que exista una caminata euleriana en un grafo G es que G debe tener a lo más dos vértices impares. Más aún, puede probarse que esta condición es también suficiente. Puesto que es sencillo calcular las valencias de los vértices de un grafo, es relativamente sencillo decidir si un grafo tendrá o no una caminata euleriana.

Resumiendo las definiciones de más arriba:

Definición 5.4.6. Un *ciclo hamiltoniano* en un grafo G es un ciclo que contiene a todos los vértices del grafo.

Una *caminata euleriana* en un grafo G es un caminata que usa todas las aristas de G exactamente una vez. Una caminata euleriana que comienza y termina en un mismo vértice se llama también *circuito euleriano*.

El siguiente teorema resume los resultados sobre caminatas eulerianas. La demostración no es demasiada complicada, pero excede los alcances de este curso.

Teorema 5.4.7. *Un grafo conexo con más de un vértice posee una caminata euleriana de v a w, con $v \neq w$ si y sólo si v y w son los únicos vértices de grado impar. Un grafo conexo con más de un vértice tiene un circuito euleriano si y sólo si todos los vértices tienen grado par.*

Observación 5.4.8. El caso de un grafo donde todas las valencias son pares se puede reducir al anterior: si deseamos una caminata euleriana que empiece y termine en v, eliminamos una arista del grafo que contenga a v, digamos la arista $\{v, w\}$ (con lo cual queda un grafo con solo dos vértices, v y w, de valencia impar), aplicamos el caso anterior, con lo cual hacemos una caminata euleriana que termina en w, y terminamos la caminata agregando la arista $\{v, w\}$.

Observación 5.4.9. Recíprocamente, el caso de un grafo G con dos valencias impares y todas las demás valencias pares se puede reducir al caso en que todas las valencias son pares. Sean p y q los dos vértices de valencia impar, entonces tenemos dos casos (1) $\{p, q\}$ es arista de G y (2) $\{p, q\}$ no es arista de G. En el caso (1), eliminamos la arista $\{p, q\}$ del grafo y nos queda un grafo con todos los vértices de valencia par. Por lo tanto, hay un circuito euleriano de p a p. Agregamos al final la arista $\{p, q\}$ y obtenemos una caminata euleriana de p a q. El caso (2) es un poco más complicado: agregamos al grafo G la arista $\{p, q\}$ y obtenemos un grafo con todos los vértices de valencia par. Luego hay un circuito euleriano de q a q. Podemos describir el circuito como

$$q, v_1, \ldots, v_k, q, p, w_1, \ldots, w_r, q.$$

A partir de este circuito podemos obtener la caminata

$$p, w_1, \ldots, w_r, q, v_1, \ldots, v_k, q,$$

que es una caminata euleriana.

Algoritmo de Hierholzer

C. Hierholzer (1840-1871) mostró, poco antes de su muerte, un algoritmo fácilmente implementable para encontrar un circuito euleriano para cualquier grafo con vértices de grado par. El algoritmo de Hierholzer fue la primera demostración del teorema 5.4.7 para el caso de grafos con vértices de grado par. Obviamente, debido a la observación 5.4.9, esto demuestra también el teorema para grafos con dos valencias impares.

El algoritmo es el siguiente:

(1) **Paso 1.** Elija cualquier vértice inicial v y haga una caminata que no repita aristas y que vuelva al vértice (de v a v).

Observar que no es posible quedarse atascado en ningún vértice que no sea v, porque el grado par de todos los vértices garantiza que, cuando se ingresa a un vértice w (distinto de v) debe haber una arista sin usar que nos permite dejar w. El recorrido formado de esta manera es un recorrido cerrado, pero puede no cubrir todos los vértices y aristas del grafo inicial.

(2) **Paso iterativo** Mientras exista un vértice u en la caminata ya realizada, pero que tenga aristas que no formen parte de la caminata, inicie otra caminata desde u hasta u siguiendo las aristas no utilizadas. Luego, inserte esta caminata a la caminata anterior para formar una caminata nueva (más larga).

Puesto que suponemos que el grafo original es conexo, repetir el paso iterativo agotará todos las aristas del grafo.

Ejemplo. Dado el grafo de de la Fig. 11, encontremos una caminata euleriana con origen en p y final en r.

Debemos primero observar que debe existir un circuito euleriano, pues $\delta(p) = 4$, $\delta(q) = 4$, $\delta(r) = 4$, $\delta(s) = 4$, $\delta(t) = 4$, y $\delta(u) = 2$, es decir todos los vértices tienen grado par.

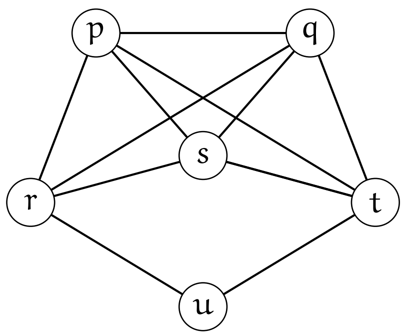

Figura 11: El gran tour, de nuevo.

Apliquemos el algoritmo de Hierholzer partiendo desde p. Una caminata posible con origen en p y que vuelva a p es

$$p, q, s, r, p.$$

Ahora elijamos q que es un vértice que pertenece a la caminata pero que tiene aristas que no son parte de la caminata. Una caminata que no toca aristas usadas y que parte de q y regresa a q es q, r, u, t, q. Insertamos en q esta caminata a la caminata anterior y obtenemos:

$$p, \mathbf{q}, \mathbf{r}, \mathbf{u}, \mathbf{t}, \mathbf{q}, s, r, p.$$

(En negrita la caminata insertada).

En el siguiente paso podemos elegir t y hacer la caminata t, p, s, t que no pasa por aristas ya utilizadas. Insertamos esta caminata en algún t de la caminata anterior y obtenemos:

$$p, q, r, u, t, p, s, t, q, s, r, p,$$

que es un circuito euleriano.

Observación. Escribiremos en pseudocódigo el algoritmo para hallar un circuito euleriano en un grafo G con n vértices de valencia par.

Tenemos una grafo G con n vértices numerados $0, 1, \ldots, n-1$, G se representa por una lista de adyacencia, es decir $G[i]$ es la lista de vértices adyacentes a i. Tomamos como comienzo del recorrido el vértice arbitrario 0.

Como ya mencionamos anteriormente, a un grafo G lo representaremos por una lista de adyacencia. Supongamos que los vértices de G son $0, \ldots, k$, entonces $G[i] = [i_0, i_1, \ldots]$ serán los vértices adyacentes al vértice i.

Caminata euleriana

```
# pre: G grafo con todos los vértices de valencia par
# post: devuelve 'circuito' una lista de vértices que forma una
#       caminata euleriana.
#       La caminata empieza en 0 (y terminará en 0).
circuito = [0]  # inicio de la caminata
libres = copiar_grafo(G) # aristas no utilizadas
while nro_aristas(libres) > 0:
    sub_cam = []
    h = 0
    while libres[h] == [] or h not in circuito:
        h = h + 1
    # h = vértice en circuito y libre[h] != [] (hay aristas libres)
    pos = circuito.index(h) # posición de la 1º ocurrencia de h
    p0 = h
    p1 = libres[h][0]
    while p1 != h: # mientras no se vuelva al origen
        sub_cam.append(p1)  # agrega p1 a sub_cam
        libres.remove({p0,p1}) # quitar arista p0, p1
        p0 = p1
        p1 = libres[p0][0]
    libres.remove({p0, h}) # quitar arista p0, h
    circuito = circuito[:pos+1] + sub_cam + circuito[pos:]
return circuito
```

En todo el programa `libres` es una lista de adyacencia que nos va dando las aristas no utilizadas. Es decir si w es un vértice `libres[w]` es una lista de los vértices u tal que la arista wu no ha sido utilizada.

§ *Ejercicios*

1) Encontrar el número de componentes de el grafo cuya lista de adyacencia es

a	b	c	d	e	f	g	h	i	j
f	c	b	h	c	a	b	d	a	a
i	g	e		g	i	c		f	f
j		g			j	e			

2) ¿Cuántas componentes conexas tiene el grafo de la fiesta de Abril (sección 5.1)?

3) Encontrar un ciclo hamiltoniano en el grafo formado por los vértices y aristas de un cubo.

4) El año que viene el Leandro y Juan desean visitar otra isla, donde los lugares interesantes y las caminos que los unen están representados por el grafo que tiene la siguiente lista de adyacencia

0	1	2	3	4	5	6	7	8
1	0	1	0	3	0	1	0	1
3	2	3	2	5	4	5	2	3
5	6	7	4		6	7	6	5
7	8		8		8		8	7.

¿Es posible encontrar rutas para Leandro y Juan que satisfagan lo pedido en el ejemplo 5.4.5?

5) Un ratón intenta comer un $3 \cdot 3 \cdot 3$ cubo de queso. Él comienza en una esquina y come un subcubo de $1 \cdot 1 \cdot 1$, para luego pasar a un subcubo adyacente. ¿Podrá el ratón terminar de comer el queso en el centro?

5.5 ÁRBOLES

Definición 5.5.1. Diremos que un grafo T es un *árbol* si cumple

T1) T es conexo y no hay ciclos en T.

Algunos árboles típicos han sido dibujados en la Fig. 12. A causa de su particular estructura y propiedades, los árboles aparecen en diversas aplicaciones de la matemática, especialmente en investigación operativa y ciencias de la computación. Comenzaremos el estudio de ellos estableciendo algunas propiedades sencillas.

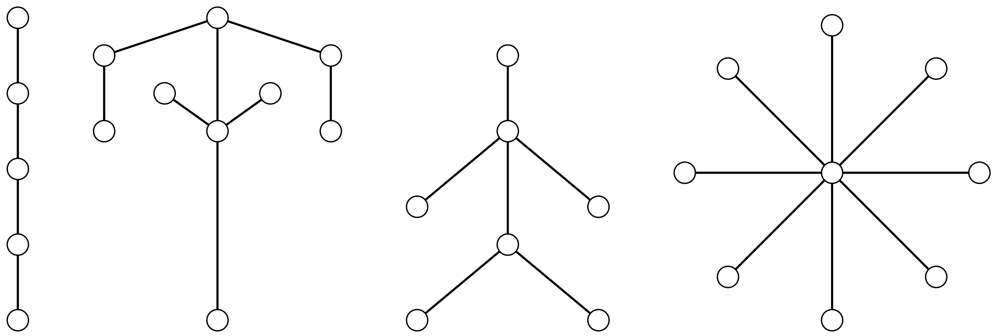

Figura 12: Algunos árboles

El siguiente lema nos resultará útil para probar una parte del teorema fundamental de esta sección.

Lema 5.5.2. *Sea* $G = (V, E)$ *un grafo conexo, entonces* $|E| \geqslant |V| - 1$.

Demostración. Como G es conexo existe una caminata que recorre todos los vértices de G:

$$v_1, v_2, \dots, v_r.$$

Renombremos los vértices de G con números naturales de tal forma que el primer vértice de la caminata sea 1, el segundo 2 y cada vez que aparece un vértice que no ha sido renombrado se le asigna el número siguiente. Luego la caminata comienza en 1 y termina en n, donde $n = |V|$. Observar que cada vez que renombramos un vértice (excepto el primero) su antecesor es menor, es decir dado i tal que $1 < i \leqslant n$ tenemos que la caminata tiene la forma

$$1, \dots, j_i, i, \dots, j_n, n$$

donde $j_i < i$, luego es claro que

$$\{j_2, 2\}, \{j_3, 3\}, \dots, \{j_n, n\}$$

forman un conjunto de $n - 1$ aristas distintas en G. □

Teorema 5.5.3. *Si* $T = (V, E)$ *es un grafo conexo con al menos dos vértices, entonces son equivalentes las siguientes propiedades*

T1) *T es un árbol.*

T2) *Para cada par* x*,* y *de vértices existe un único camino en* T *de* x *a* y*.*

T3) *El grafo obtenido de* T *removiendo alguna arista tiene dos componentes, cada una de las cuales es un árbol.*

T4) $|E| = |V| - 1$.

Demostración.
(T1 ⇒ T2) Puesto que T es conexo, existe un camino de x a y, digamos

$$x = v_0, v_1, \ldots, v_r = y.$$

Si existiera otro camino, digamos

$$x = u_0, u_1, \ldots, u_s = y,$$

consideremos i el más pequeño subíndice para el cual se cumple que $u_{i+1} \neq v_{i+1}$ Fig. 13.

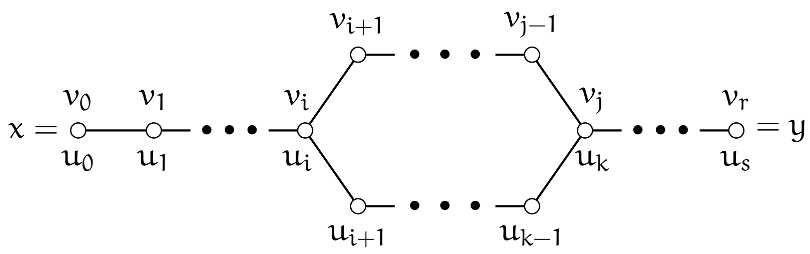

Figura 13: Dos caminos diferentes determinan un ciclo

Puesto que ambos caminos finalizan en y ellos se encontrarán de nuevo, y entonces podemos definir j como el más pequeño subíndice tal que

$$j > i \quad \text{y} \quad v_j = u_k \quad \text{para algún } k.$$

Entonces $v_i, v_{i+1}, \ldots, v_j, u_{k-1}, u_{k-2}, \ldots, u_{i+1}, v_i$ es un ciclo en T, y esto contradice a las hipótesis. Por consiguiente solo existe un camino en T de x a y.
(T2 ⇒ T3) Supongamos que uv es una arista en T, y sea $S = (V, E')$ el grafo con el mismo conjunto de vértices que T y con el conjunto de aristas $E' = E - uv$. Sea V_1 el conjunto de los vértices x de T para los cuales existe un único camino en T de x a v que pasa por u. Claramente, este camino debe finalizar con la arista uv, pues sino T tendría un ciclo. Sea V_2 el complemento de V_1 en V.

Cada vértice en V_1 se une por un camino en S a u, y cada vértice en V_2 se une por un camino en S a v, pero no existe camino de u a v en S. Se sigue entonces que V_1 y V_2 son las dos componentes del conjunto de vértices de S. Cada componente es conexa (por definición), y no contiene ciclos, pues sino habría ciclos en T. Es decir que las dos componentes son árboles.
(T3 ⇒ T4) El resultado es cierto cuando $|V| = 1$, puesto que el árbol de un vértice no tiene aristas.

Supongamos que es cierto para árboles con k o menos vértices. Sea T un árbol con $|V| = k + 1$, y sea uv una arista en T. Si $T_1 = (V_1, E_1)$ y $T_2 = (V_2, E_2)$ son los árboles que se obtienen removiendo uv en T, tenemos que

$$|V_1| + |V_2| = |V|, \qquad |E_1| + |E_2| = |E| - 1.$$

Aplicando la hipótesis inductiva a T_1 y T_2 obtenemos

$$|E| = |E_1| + |E_2| + 1 = |V_1| - 1 + |V_2| - 1 + 1 = |V| - 1,$$

como nosotros deseábamos. Por consiguiente el resultado es cierto para todos lo enteros positivos.
(T4 $\Rightarrow$ T1) Supongamos que T satisface (T4) pero no es árbol. Por lo tanto T tiene al menos un ciclo. Si eliminamos una arista del ciclo, el grafo sigue siendo conexo, pero con una arista menos (y la misma cantidad de vértices), es decir obtenemos un grafo $G' = (V', E')$ conexo y con $|E'| = |V'| - 2$. Pero esto contradice el resultado obtenido en el lema 5.5.2. □

Las propiedades (T2)-(T3)-(T4) nos dan maneras alternativas de definir árboles. Por ejemplo la propiedad (T2) puede ser considerada como la propiedad que define un árbol, en vez de (T1).
Observación. El teorema anterior nos muestra un recurso muy usado para probar que una cierta cantidad de afirmaciones son equivalentes. En el caso de 3 afirmaciones P, Q, R, uno debería probar

$$P \Leftrightarrow Q, \qquad P \Leftrightarrow R, \qquad Q \Leftrightarrow R,$$

y eso nos garantizaría la equivalencia entre P, Q y R. Si embargo, podemos ahorrar trabajo demostrando solamente

$$P \Rightarrow Q, \qquad Q \Rightarrow R, \qquad R \Rightarrow P,$$

pues si queremos probar, por ejemplo $P \Leftrightarrow Q$, esto es equivalente a probar $P \Rightarrow Q$ y $Q \Rightarrow P$. La parte $P \Rightarrow Q$ ya ha sido probada y $Q \Rightarrow P$ se deduce de $Q \Rightarrow R$, $R \Rightarrow P$ y la propiedad transitiva de $\Rightarrow$.
Para el caso de cuatro proposiciones P_1, P_2, P_3, P_4 la economía de demostraciones es aún más drástica: para probar todas las equivalencias posibles de cuatro afirmaciones hacen falta 12 demostraciones (seis de ida y seis vuelta), pero alcanza haciendo sólo 4 demostraciones:

$$P_1 \Rightarrow P_2, \qquad P_2 \Rightarrow P_3, \qquad P_3 \Rightarrow P_4, \qquad P_4 \Rightarrow P_1.$$

§ *Ejercicios*

1) Hay seis diferentes (es decir, no isomorfos entre si) árboles con seis vértices: hacer un dibujo de ellos.

2) Sea $T = (V, E)$ un árbol con $|V| \geqslant 2$. Usando la propiedad (T4) y el teorema 5.3.1 probar que T tiene al menos dos vértices con valencia 1.

3) Una *foresta* es un grafo que satisface que no contiene ciclos pero no necesariamente es conexo. Probar que si $F = (V, E)$ es una foresta con c componentes entonces

$$|E| = |V| - c.$$

5.6 COLOREO DE LOS VÉRTICES DE UN GRAFO

Un problema que se nos presenta frecuentemente en la vida moderna es aquel de confeccionar un horario para un conjunto de eventos de tal manera de evitar interferencias. Consideremos ahora un caso muy simple, que nos servirá de ejemplo para mostrar como la teoría de grafos puede ayudar al estudio de este problema.

Supongamos que deseamos hacer un horario con seis cursos de una hora, $v_1, v_2, v_3, v_4, v_5, v_6$. Entre la audiencia potencial hay gente que desea asistir a v_1 y v_2, v_1 y v_4, v_3 y v_5, v_2 y v_6, v_4 y v_5, v_5 y v_6 y v_1 y v_6. ¿Cuántas horas son necesarias para poder confeccionar un horario en el cual no haya interferencias?

Podemos representar la situación por un grafo Fig. 14. Los vértices corresponden a las seis clases, y las aristas indican las interferencias potenciales.

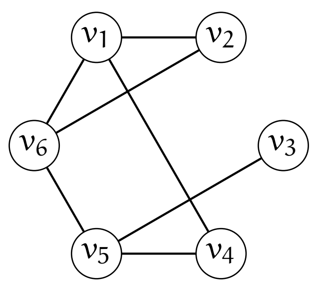

Figura 14: El grafo para un problema de horarios

Un horario el cual cumple con la condición de evitar interferencias es el siguiente:

Hora 1	Hora 2	Hora 3	Hora 4
v_1 y v_3	v_2 y v_4	v_5	v_6

En términos matemáticos, tenemos una partición del conjuntos de vértices en cuatro partes, con la propiedad que ninguna parte contiene un par de vértices adyacentes del grafo. Un descripción más gráfica utiliza la función

$$c : \{v_1, v_2, v_3, v_4, v_5, v_6\} \to \{1, 2, 3, 4\}$$

la cual asigna cada vértice (curso) a la hora que le corresponde. Usualmente, nosotros hablamos de colores asignados a los vértices, en vez de horas, pero claramente la naturaleza exacta de los objetos $1, 2, 3, 4$ no es importante. Podemos usar el nombre de colores reales, rojo, verde, azul , amarillo, o podemos hablar del color 1, color 2, etc. Lo importante es que los vértices que son adyacentes en el grafo deben tener diferentes colores.

Definición 5.6.1. Una *coloración de vértices* de un grafo $G = (V, E)$ es una función $c : V \to \mathbb{N}$ con la siguiente propiedad:

$$c(x) \neq c(y) \quad \text{si} \quad \{x, y\} \in E.$$

El *número cromático* de G, denotado $\chi(G)$, se define como el mínimo entero k para el cual existe una coloración de vértices de G usando k-colores. En otra palabras, $\chi(G) = k$ si y sólo si existe una coloración de vértices c la cual es una función de V a $\mathbb{N}_k$, y k es el mínimo entero con esta propiedad.

Volviendo al ejemplo de la Fig. 14, vemos que nuestro primer intento de horario es equivalente a una coloración de vértices con cuatro colores. El mínimo número de horas necesarias será el número cromático del grafo, y la pregunta es ahora si este número es cuatro o menor que cuatro. Un rápido intento con tres colores nos da la solución de este problema:

Color 1	Color 2	Color 3
v_1	v_2 y v_5	v_3, v_4 y v_6.

Más aún, hacen falta por lo menos tres colores, puesto que v_1, v_2, y v_6 son mutuamente adyacentes y por lo tanto deben tener diferentes colores. Luego concluimos que el número cromático del grafo es 3.

En general, para probar que el número cromático de un grafo dado es k, debemos hacer dos cosas:

a) encontrar una coloración de vértices usando k colores;

b) probar que ninguna coloración de vértices usa menos de k colores.

§ *Ejercicios*

1) Encontrar el número cromático de los siguientes grafos:
   a) un grafo completo K_n;
   b) un grafo cíclico C_{2r} con un número par de vértices;
   c) un grafo cíclico C_{2r+1} con un número impar de vértices.

2) Determinar los números cromáticos de los grafos descritos en la Fig. 15.

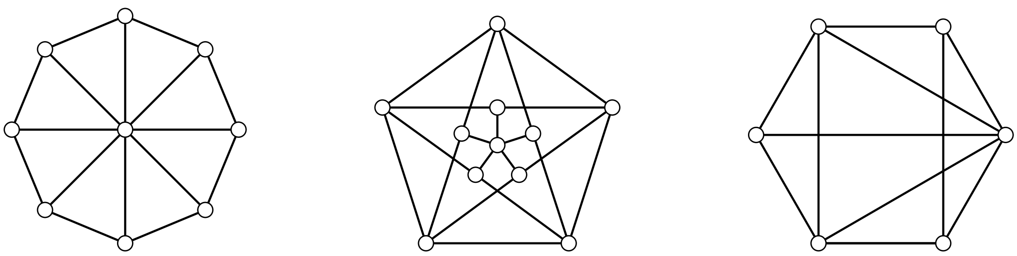

Figura 15: Encontrar el número cromático

3) Describir todos los grafos G tales que $\chi(G) = 1$.

5.7 EL ALGORITMO GREEDY PARA COLORACIÓN DE VÉRTICES

Es bastante difícil encontrar el número cromático de un grafo dado. En realidad, no se conoce ningún algoritmo para este problema que trabaje en “tiempo polinomial”, y la mayoría de la gente cree que tal algoritmo no existe. Sin embargo hay un método simple de hacer una coloración cromática usando un “razonable” número de colores.

El método consiste en asignar los colores de los vértices en orden, de tal manera que cada vértice recibe el primer color que no haya sido ya asignado a alguno de sus vecinos. En este algoritmo insistimos en hacer la mejor elección que podemos en cada paso, sin mirar más allá para ver si esta elección nos traerá problemas luego. Un algoritmo de esta clase se llama a menudo un *algoritmo greedy (goloso)*.

El algoritmo greedy para coloración de vértices es fácil de programar. Supóngase que hemos dado a los vértices algún orden $v_0, v_1, \ldots, v_n$. Asignemos el color 0 a v_0 y luego le vamos asignando un color a los subsiguientes vértices: para cada v_i ($1 \leqslant i \leqslant n$) formamos el conjunto S de colores asignados a los vértices v_j ($0 \leqslant j < i$) que son adyacentes a v_i, y le damos a v_i el primer color que no está en S. (En la práctica, pueden ser usados métodos más sofisticados de manejar los datos.)

ALGORITMO GREEDY PARA COLORACIÓN DE VÉRTICES

```
# pre: 0,...,n los vértices de un grafo G
# post: devuelve v[0],...,v[n] una coloración de G
color = []  #  color[j] = c dirá que el color de j es c.
for i = 0 to n:
    S = []  # S conjunto de colores asignados a los vértices j
            # (1 <= j <i) que son adyacentes a i (comienza vacío)
    for j = 0 to i-1:
        if j es adyacente a i:
           S.append(color[j])  # agrega el color de j a  S
    k = 0
    while k in S:
        k = k + 1
    color.append(k) # Asigna el color k a i, donde k es el primer
                    # color que no esta en S.
```

Debido a que la estrategia greedy es corta de vista, el número de colores que usará será normalmente más grande que le mínimo posible. Por ejemplo, el algoritmo greedy aplicado en el grafo de Fig. 14 da precisamente le coloración de vértices con cuatro colores que fue propuesta anteriormente, luego encontramos otra coloración con tres colores. Por supuesto todo depende del orden que se elige inicialmente para los vértices. Es bastante fácil ver que si se elige el orden correcto, entonces el algoritmo greedy nos da la mejor coloración posible (ejercicio 5.7-(2)). Pero hay $n!$ órdenes posibles, y si tuviéramos que controlar cada uno de ellos, el algoritmo requeriría "tiempo exponencial".

Más allá de esto, el algoritmo greedy es útil tanto en la teoría como en la práctica. Probaremos ahora dos teoremas por medio de la estrategia greedy.

Teorema 5.7.1. *Si G es un grafo con valencia máxima k, entonces*

a) $\chi(G) \leqslant k+1$,

b) *Si G es conexo y no regular*, $\chi(G) \leqslant k$.

Demostración.

a) Sea $v_1, v_2, \ldots, v_n$ un ordenamiento de los vértices de G. Cada vértice tiene a lo más k vecinos, y por consiguiente el conjunto S de los colores asignados por el algoritmo greedy a los vértices v_j que son adyacentes a v_i $(1 \leqslant j < i)$ tiene como máximo cardinal k. Por consiguiente al menos uno de los colores $1, 2, \ldots, k+1$ no está en S, y el algoritmo greedy asigna entonces el primero de estos a v_i.

b) Para probar esta parte debemos elegir un orden especial de los vértices, comenzando con v_n y yendo hacia atrás. Puesto que G tiene valencia máxima k y es no regular, existe al menos un vértice G cuya valencia es menor que k: llamémoslo v_n. Listemos los vecinos de v_n como $v_{n-1}, v_{n-2}, \ldots, v_{n-r}$; hay a lo más $k-1$ de ellos. A continuación listemos los vecinos de v_{n-1} (excepto v_n y sus vecinos), y observemos que como la valencia es a lo más k hay a lo más $k-1$ de estos vértices. A continuación listemos los vecinos de v_{n-2} que no hayan sido listados antes, y así siguiendo. Puesto que G es conexo, en determinado momento podremos listar todos los vértices de G. Más aún, el método de construcción asegura que cada vértice es adyacente a lo más a $k-1$ de sus predecesores en el orden $v_1, v_2, \ldots, v_n$.

Usando el mismo argumento que en la parte *a)* (pero para este orden) se sigue que el algoritmo greedy requerirá a lo más k colores. Luego $\chi(G) \leqslant k$. □

La parte *b)* del teorema es falsa si permitimos que G sea regular. El lector que haya respondido correctamente al ejercicio *1* de la sección 5.6 será capaz de dar dos ejemplos de este hecho: los grafos cíclicos de longitud impar tienen grado 2 y número cromático 3 y los grafos completos de $k+1$ vértices tienen grado k y número cromático $k+1$ colores. Si embargo, puede ser demostrado que estos son los únicos contraejemplos.

Otra consecuencia útil del algoritmo greedy se refiere a grafos G son $\chi(G) = 2$. Para tales grafos, los conjuntos V_1 y V_2 de vértices de colores 1 y 2 respectivamente, forman una partición de V, con la propiedad que cada arista tiene un vértice en V_1 y el otro en V_2. Por esta razón, cuando $\chi(G) = 2$, diremos que G es *bipartito*. Una coloración de vértices con dos colores de un cubo se ilustra en la Fig. 16, junto a un dibujo alternativo que enfatiza la naturaleza bipartita del grafo. Usualmente usaremos esta clase de dibujo cuando trabajemos con grafos bipartitos.

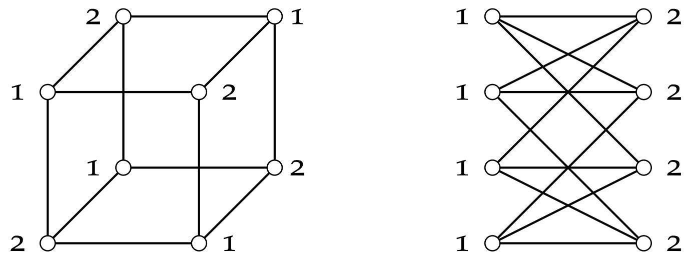

Figura 16: El cubo es un grafo bipartito

Teorema 5.7.2. *Un grafo es bipartito si y sólo si no contiene ciclos de longitud impar.*

Demostración. Si hay un ciclo de longitud impar, entonces se requieren tres colores, solamente para colorear este ciclo, y el número cromático del grafo es por ende al menos tres. Luego si el grafo es bipartito, no puede tener ciclos de longitud impar.

Recíprocamente, supongamos que G es un grafo sin ciclos de longitud impar. Construiremos un orden de G para el cual el algoritmo greedy producirá una coloración de vértices con dos colores. Elijamos cualquier vértice y llamémoslo v_1; diremos que v_1 esta en el *nivel* 0. A continuación, listemos la lista de vecinos de v_1 (excepto v_1), llamémoslos $v_2, v_3, \dots, v_r$; diremos que estos vértices están en el *nivel 2*. Continuando de esta manera, definimos el *nivel* l como todos aquellos vértices adyacentes a los del *nivel* $l-1$, exceptuando aquellos previamente listados en el *nivel* $l-2$. Cuando ningún nuevo vértice puede ser agregado de esta forma, obtenemos la componente G_0 de G (si G es conexo $G_0 = G$).

El hecho crucial producido por este orden es que un vértice del nivel l solo puede ser adyacente a vértices de los niveles $l-1$ y $l+1$, y no a vértices del mismo nivel. Supongamos que x e y son vértices en el mismo nivel; entonces ellos son unidos por caminos de igual longitud m a algún vértice z de un nivel anterior, y los caminos pueden ser elegidos de tal manera que z sea el único vértice común Fig. 17. Si x e y fueran adyacentes, habría un ciclo de longitud $2m+1$, lo cual contradice la hipótesis.

Se deduce entonces que el algoritmo greedy asigna el color 1 a los vértices en el nivel $0, 2, 4, \dots$, y el color 2 a los vértices en los niveles $1, 3, 5, \dots$. Por consiguiente $\chi(G_0) = 2$. Repitiendo el mismo argumento para cada componente de G obtenemos el resultado deseado. $\square$

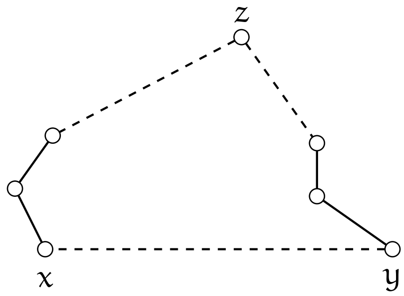

Figura 17: Vértices adyacentes en el mismo nivel inducen un ciclo impar

§ *Ejercicios*

1) Encontrar órdenes de los vértices del grafo del cubo Fig. 16 para los cuales el algoritmo greedy requiera $2, 3$ y 4 colores respectivamente.

2) Probar que para cualquier grafo G existe un orden de los vértices para el cual el algoritmo greedy requiera $\chi(G)$ colores. [Ayuda: use un coloreado de vértices de $\chi(G)$ colores para definir el orden.]

3) Denotar $e_i(G)$ el número de vértices del grafo G cuya valencia es estrictamente mayor que i. Usar el algoritmo greedy para probar que si $e_i(G) \leqslant i+1$ para algún i, entonces $\chi(G) \leqslant i+1$.

4) El grafo M_r ($r \geqslant 2$) se obtiene a partir del grafo cíclico C_{2r} añadiendo aristas extras que unen los vértices opuestos. Probar que

   a) M_r es bipartito cuando r es impar,

   b) $\chi(M_r) = 3$ cuando r es par y $r \neq 2$,

   c) $\chi(M_2) = 4$.