

Grafos

AMD – Grado en Ingeniería Informática

Objetivos

Al finalizar este tema tendréis que:

- Conocer la terminología básica de la teoría de grafos.
- Pasar de una representación gráfica de un grafo a su expresión matricial y viceversa.
- Conocer los siguientes tipos de caminos: abiertos, cerrados, simples y eulerianos.
- Saber si un grafo es Euleriano.
- Calcular el árbol generador de peso mínimo de un grafo conexo.
- Calcular el camino más corto entre dos vértices.

Primeras definiciones

Un **grafo** es un conjunto de **vértices** unidos por **aristas**. Gráficamente, los vértices son puntos y las aristas son líneas que los unen.

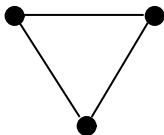


Figura: Grafo con 3 vértices y 3 aristas

Sirven para representar varios tipos de problemas y sin duda es una herramienta que utilizaréis a lo largo de vuestros estudios. Aquí os daremos unas nociones básicas y si da tiempo, veremos alguna aplicación en el aula.

Primeras definiciones

Hay mucha clase de grafos, mirad

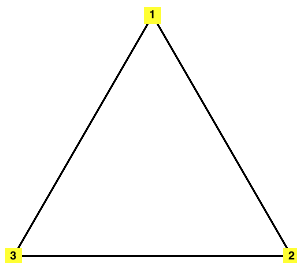
http://es.wikipedia.org/wiki/Teoria_de_grafos,

pero sólo vamos a trabajar con los llamados **simples no dirigidos**: las aristas no son flechas y dos vértices se unen a lo sumo por una arista (no tienen que estar unidos).

Si queremos trabajar con un grafo, conviene etiquetar los vértices con números, letras, colores, etc; e identificar las aristas con los dos vértices que unen. Por ejemplo el grafo anterior se puede ver como

$$V = \{1, 2, 3\}, E = \{\{1, 2\}, \{2, 3\}, \{1, 3\}\}$$

donde el conjunto de vértices es V y el de aristas E ,



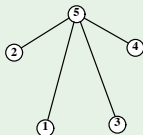
Primeras definiciones

Definición

El **grado** de un vértice es el número de aristas de las que es extremo. Se dice que un vértice es 'par' o 'impar' según lo sea su grado.

Ejemplo

En este grafo, el vértice 5 tiene grado cuatro, y los demás grado uno.



El siguiente resultado relaciona los grados con el n° de aristas

Lema del apretón de manos

La suma de todos los grados es igual al doble del n° de aristas.

Matriz de adyacencia

Toda la información de un grafo, es decir, los vértices y cuáles de éstos están unidos por aristas, puede representarse en una matriz: la de adyacencia.

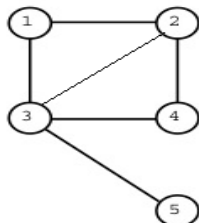
Definición

Sea G un grafo simple con n vértices. La matriz de adyacencia de G es una matriz cuadrada simétrica de tamaño $n \times n$, que denotaremos A , tal que:

- $a(i, j) = 1$ si los vértices i y j están unidos,
- $a(i, j) = 0$ si no.

Para hallar la matriz de adyacencia necesitamos que los vértices estén etiquetados.

Ejemplo



$$A = \begin{matrix} & \begin{matrix} v_1 & v_2 & v_3 & v_4 & v_5 \end{matrix} \\ \begin{matrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{matrix} & \begin{pmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{matrix}$$

Matriz de adyacencia del grafo de la izquierda.

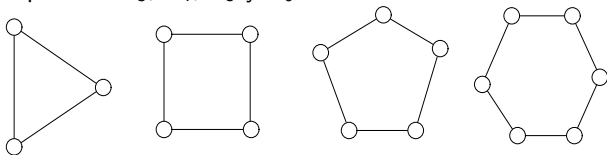
Matriz de adyacencia

Observad que dada una matriz de adyacencia, podemos ver el grafo. ¿Sabrías dibujar los grafos correspondientes a estas matrices?

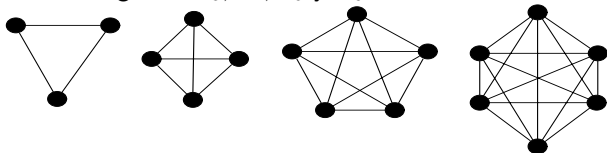
$$A_1 = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}, A_2 = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}, A_3 = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \end{pmatrix}$$

Ejemplos de grafos

Grafo ciclo Grafo que tiene el mismo número de vértices (≥ 3) que de aristas, las cuales van uniendo todos los vértices formando un ciclo. El grafo ciclo con n vértices se denota C_n . En la siguiente figura aparecen C_3 , C_4 , C_5 y C_6 .



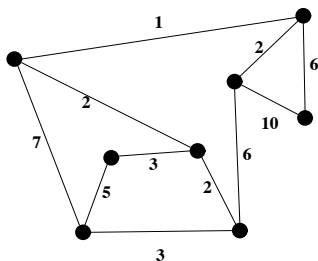
Grafo completo Grafo en el que cada vértice está unido a todos los demás. Un grafo completo con n vértices se denota K_n . A continuación pueden verse los grafos K_3 , K_4 , K_5 y K_6 .



Fijaros que el grado de cada vértice es $n - 1$.

Ejemplos de grafos

Grafo ponderado Si a cada arista de un grafo se le asocia un número positivo, llamado “peso”, diremos que el grafo es ponderado. Por ejemplo,



Árbol Grafo en el que dos vértices distintos están conectados por un único camino.

Ejemplos de grafos

Grafos bipartidos El conjunto de vértices V se divide en dos subconjuntos disjuntos, denotados por A y B , tal que toda arista tiene como extremos un vértice de A y otro de B .

Dentro de los grafos bipartidos, se encuentran los **grafo bipartidos completos**, en los cuales cada vértice de A está conectado a cada vértice de B . Se denotan $K_{m,n}$ donde m, n es el orden de cada subconjunto disjunto de vértices y $m \leq n$. Observad que $K_{m,n}$ tiene $m + n$ vértices y mn aristas.

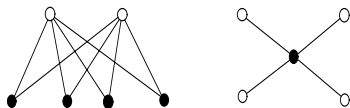


Figura: $K_{2,4}$ y $K_{1,4}$

Caminos y conexión

Uno de los problemas en teoría de grafos es ver si entre dos vértices hay un camino.

Definición

Un **camino** C entre dos vértices v_1 y v_n es una sucesión de aristas de la forma

$$\{v_1, v_2\}, \{v_2, v_3\}, \dots, \{v_{n-1}, v_n\}$$

que une los vértices v_1 y v_n .

Dado un camino entre v_1 y v_n , entonces

- el número de aristas que recorre se denomina **longitud** del camino,
- es **cerrado** si $v_1 = v_n$, y **abierto** si lo contrario;
- es **simple** si no pasa dos veces por el mismo vértice;
- es un **ciclo** si es simple y cerrado, es decir, cerrado donde el único vértice que se repite es el inicial como punto de llegada.

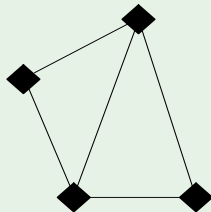
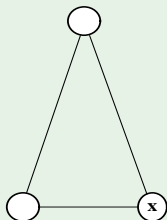
Caminos y conexión

Definición

Un grafo G es **conexo** si cada par de vértices está unido al menos por un camino.

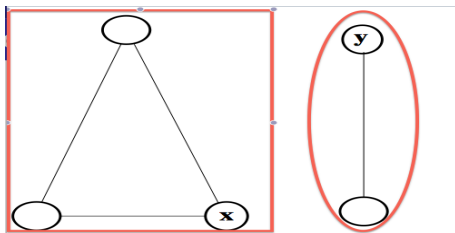
Ejemplo

En el grafo de la izquierda, con vértices blancos, no hay camino entre los vértices x e y y no es conexo. En cambio, el grafo de la derecha, con vértices negros, sí lo es.



Caminos y conexión

Un grafo que no es conexo es **disconexo**. Cada grafo disconexo se puede desglosar en un número de subgrafos conexos, cada uno de los cuales se dice que es una **componente conexa del grafo**. En el ejemplo anterior, el grafo disconexo se compone de dos componentes conexas,



Una arista en un grafo conexo se llama de **separación** si al eliminarla, el grafo es disconexo.

Caminos y matriz de adyacencia

Sea G un grafo con $n > 1$ vértices y A su matriz de adyacencia, entonces las potencias de dicha matriz, A^k , indican el número de caminos de longitud k entre dos vértices. Es decir,

Teorema

La entrada en la fila i y la columna j de la matrix A^k , denotada por $a_{ij}^{(k)}$, es igual al número de caminos de longitud k con extremos v_i y v_j .

Esto nos da el siguiente método para ver si un grafo es conexo:

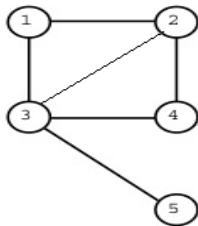
Algoritmo para ver si un grafo es conexo

1. Se halla la matriz de adyacencia y se calcula hasta A^{n-1} .
2. Se calcula la suma de las potencias de A hasta A^{n-1} , es decir, $A + A^2 + A^3 + \dots + A^{n-1}$.
3. El grafo es conexo si y sólo si ninguna entrada de esta matriz suma es 0.

Obviamente con grafos pequeños no merece la pena utilizar este algoritmo, pero aún así veamos que significa el teorema y el algoritmo con un ejemplo.

Caminos y matriz de adyacencia. Ejemplo

Consideremos el ejemplo de la transparencia 9:



$$A = \begin{matrix} & v_1 & v_2 & v_3 & v_4 & v_5 \\ \begin{matrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{matrix} & \begin{pmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{matrix}$$

Matriz de adyacencia del grafo de la izquierda

Las potencias de la matriz son la siguientes y veamos entre todos qué significan:

$$A^2 = \begin{pmatrix} 2 & 1 & 1 & 2 & 1 \\ 1 & 3 & 2 & 1 & 1 \\ 1 & 2 & 4 & 1 & 0 \\ 2 & 1 & 1 & 2 & 1 \\ 1 & 1 & 0 & 1 & 1 \end{pmatrix}, A^3 = \begin{pmatrix} 2 & 5 & 6 & 2 & 1 \\ 5 & 4 & 6 & 5 & 2 \\ 6 & 6 & 4 & 6 & 4 \\ 2 & 5 & 6 & 2 & 1 \\ 1 & 2 & 4 & 1 & 0 \end{pmatrix}, A^4 = \begin{pmatrix} 11 & 10 & 10 & 11 & 6 \\ 10 & 16 & 16 & 10 & 6 \\ 10 & 16 & 22 & 10 & 4 \\ 11 & 10 & 10 & 11 & 6 \\ 6 & 6 & 4 & 6 & 4 \end{pmatrix}$$

Grafos Eulerianos

El “problema de los puentes de Königsberg” originó el concepto de grafo euleriano.

Definición

Un grafo G diremos que es **Euleriano** si existe un circuito (camino cerrado) que recorre todas las aristas sin repetir ninguna. A dicho circuito lo llamaremos **circuito euleriano**.

Observad que un camino que recorre todas las aristas, recorre todos los vértices. El recíproco no es cierto.

Ejemplo

Cualquier grafo ciclo es euleriano.

¿Se os ocurren más ejemplos?

Grafos Eulerianos

Estudiar si un grafo es euleriano o no es muy sencillo debido al siguiente resultado.

Propiedad de los grafos eulerianos

Un grafo conexo es euleriano si y sólo si todo vértice tiene grado par.

Este resultado nos da más ejemplos de grafos eulerianos.

Ejemplo

Aplicando el anterior resultado, sabemos que:

- los grafos completos K_n son eulerianos cuando $n - 1$ es par,
- los grafos completos bipartidos $K_{n,m}$ son eulerianos cuando n y m son ambos pares.

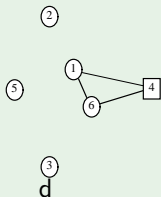
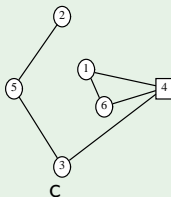
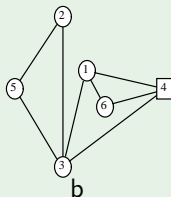
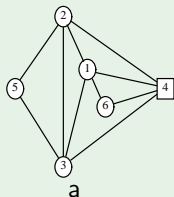
Para encontrar un circuito euleriano podemos utilizar el **algoritmo de Fleury**, descrito en la página siguiente.

Pasos en el algoritmo de Fleury y ejemplo

- 1 comenzamos eligiendo un vértice del grafo,
- 2 vamos recorriendo el grafo, eligiendo aristas que no se hayan recorrido,
- 3 en cada paso, sólo se recorre una arista de separación del subgrafo formado por las aristas que quedan sin elegir si no queda otra alternativa.

Ejemplo

En a) tenemos el grafo original. Elegimos el vértice 4 y la arista $\{4, 2\}$, seguida de la $\{2, 1\}$, quedando b). Seguidamente elegimos la arista $\{1, 3\}$, seguida de $\{3, 2\}$, quedando c). En el próximo paso, se cogen aristas de separación al no tener otra elección, pasando por $\{2, 5\}$, $\{5, 3\}$ y $\{3, 4\}$. En d) sólo nos queda un K_3 , recorriéndolo con $\{4, 6\}$, $\{6, 1\}$, y $\{1, 4\}$. El circuito euleriano encontrado es $\{4, 2\}\{2, 1\}\{1, 3\}\{3, 2\}\{2, 5\}\{5, 3\}\{3, 4\}\{4, 6\}\{6, 1\}\{1, 4\}$



¿Qué puede ocurrir si un grafo no es euleriano?

Cabe preguntarse si al menos hay un camino abierto euleriano.

Definición

Un camino abierto euleriano en un grafo es un camino abierto que recorre todas las aristas del grafo tan sólo una vez

Gracias al siguiente resultado sabemos si existe un camino abierto euleriano o no.

Propiedad

Un grafo conexo tiene un camino abierto euleriano si y sólo si tiene exactamente dos vértices de grado impar

Esto nos lleva al siguiente algoritmo para encontrar un camino abierto euleriano en un grafo conexo:

- 1 se añade una arista que una lo dos vértices de grado impar, así pasarán a tener grado par;
- 2 se busca el circuito euleriano del grafo resultante;
- 3 se borra de dicho circuito la arista que hemos añadido.

Árboles

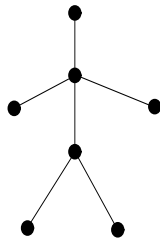
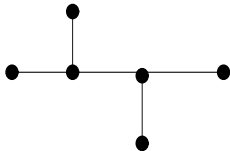
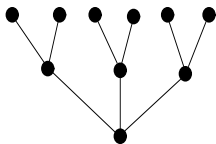
Definición

Un árbol es un grafo conexo y sin ciclos.

De esta definición podemos deducir que:

- Entre dos vértices diferentes, hay un único camino. Si no, obtendríamos un ciclo.
- El número de vértices es el número de aristas + 1.

Los siguientes tres grafos son árboles:



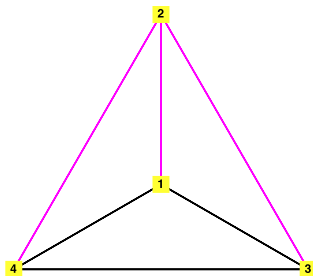
Árboles generadores

Un grafo conexo que no es árbol, tiene ciclos. Si suprimimos algunas aristas para eliminar los ciclos, podremos obtener diferentes árboles “escondidos” en el grafo.

Definición

Sea G un grafo. Cualquier subgrafo de G que sea un árbol y tenga todos los vértices de G se dice árbol generador de G .

Encontrar un árbol generador es sencillo; por ejemplo, las aristas en magenta definen un árbol generador de K_4 :



Árboles generadores

Sin embargo, supongamos que tenemos un grafo conexo ponderado. Es decir, a cada arista se le asigna un peso. Ahora la pregunta formulada es: **¿Sabrías encontrar un árbol generador de peso mínimo?** Se sobreentiende que el peso de un árbol es la suma de los pesos de sus aristas.

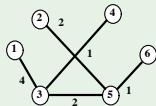
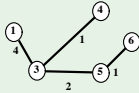
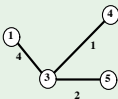
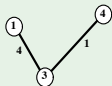
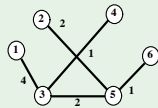
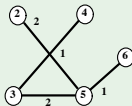
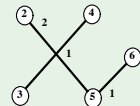
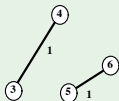
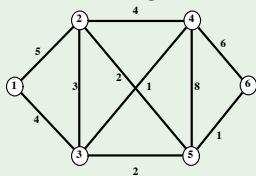
Para hallar un árbol de peso mínimo, llamado T , existen 2 algoritmos:

Algoritmos para encontrar árboles generadores de peso mínimo

- *Kruskal*: Se van eligiendo las aristas de menor peso, de tal forma que NO se formen ciclos, hasta conseguir T .
- *Prim*: Se elige un vértice de forma arbitraria, siendo el primer vértice de nuestro árbol T . Sucesivamente, en cada paso se localiza la arista de peso mínimo que conecte al árbol T con un vértice que aún no está en T , de forma que sigamos teniendo un árbol. Se añade dicha arista, dicho vértice y se continúa hasta tener todos los vértices.

Ejemplo

En este ejemplo, se aprecia la diferencia de los dos métodos. Primero utilizamos el algoritmo de Kruskal y después, partiendo del vértice 1, aplicamos el algoritmo de Prim. El grafo tiene al menos un árbol generador de peso mínimo 8.



El problema del camino más corto

Ahora tenemos un grafo conexo ponderado y dados dos vértices, la pregunta que nos hacemos es: **¿Cuál es el camino más corto entre ambos?** o lo que es lo mismo: ¿Cuál es el árbol de menos peso entre ambos? Hemos suprimido la palabra “generador” porque ahora no se tienen que coger todos los vértices del grafo.

El algoritmo que vamos a describir para hallar dicho camino se denomina **Algoritmo de Dijkstra**.

Algoritmo de Dijkstra

Tenemos n vértices, $\{v_1, \dots, v_n\}$, y queremos hallar el **camino más corto** entre v_1 y v_n .

Para encontrarlo, vamos generando los pesos de los caminos mínimos desde v_1 hasta los demás vértices, y estos pesos se van almacenando en una tabla, en un campo llamado *Dis*. La tabla tiene la siguiente estructura:

Iteración	Dis.				u	S
	$p(v_1)$	$p(v_2)$	\dots	$p(v_n)$		

En la columna S vamos añadiendo aristas, que forman al final un árbol de v_1 a v_n . En la columna u los vértices que van definiendo el árbol. Finalmente, el camino de dicho árbol entre v_1 y v_n será la solución del problema.

Algoritmo de Dijkstra

La tabla se hace añadiendo filas, de izquierda a derecha, del siguiente modo:

- $p(v_1) = 0$ siempre, pues es la distancia más corta de v_1 a sí mismo.
- En la iteración 1, para cada vértice v_j , si es adyacente a v_1 , entonces $p(v_j)$ es el peso de la arista que los une; si no, $p(v_j) = \infty$.

Así, el mínimo de $\{p(v_2), \dots, p(v_n)\}$ nos proporciona el vértice más próximo a v_1 . Supongamos que lo llamamos x . Entonces, a la columna u se añade el vértice x ; a la columna S la arista $\{v_1, x\}$ y el valor $p(x)$ queda fijo.

- En la iteración 2, para cada vértice v_j que no forma parte del árbol, ($v_j \neq v_1, v_j \neq x$) se actualiza $p(v_j)$, pasando ser igual a:

$$\text{mín}\{p(v_j), p(x) + p_{\{x, v_j\}}\}, \text{ si } v_j \text{ es adyacente a } x$$

si no, el valor $p(v_j)$ es el mismo de la iteración anterior,

obteniendo el peso del camino mínimo de longitud \leq que 2 desde v_1 hasta v_j .

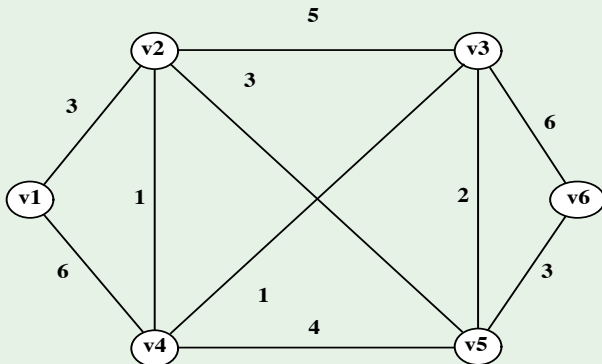
El mínimo del nuevo listado $\{p(v_2), \dots, p(v_n)\}$ proporciona un nuevo vértice.

Si lo llamamos z , entonces $p(z)$ queda fijo, siendo el peso del camino mínimo entre v_1 y z , el vértice z se añade a la columna u , la arista $\{a, z\}$ a la columna S , donde a es el vértice del árbol que estamos construyendo, utilizado para obtener z (en la iteración 2, $a = v_1$ o $a = x$).

- Así sucesivamente, hasta añadir v_n a la columna u .

Ejemplo

Tenemos el siguiente grafo



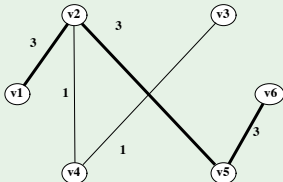
y queremos hallar el camino más corto de v_1 hasta v_6 .

Ejemplo

Con el algoritmo de Dijkstra, obtenemos la siguiente tabla:

Iteración	Dis.						u	S
	$p(v_1)$	$p(v_2)$	$p(v_3)$	$p(v_4)$	$p(v_5)$	$p(v_6)$		
1	0	<u>3</u>	∞	6	∞	∞	v_2	$\{v_1, v_2\}$
2			8	<u>4</u>	6	∞	v_4	$\{v_2, v_4\}$,
3			<u>5</u>		6	∞	v_3	$\{v_4, v_3\}$
4					<u>6</u>	11	v_5	$\{v_2, v_5\}$
5						<u>9</u>	v_6	$\{v_5, v_6\}$

tal que las aristas en S forman el árbol de la siguiente figura, donde el camino marcado es el mínimo entre v_1 y v_6 , de longitud 9.



Ideas para entregable de subir nota

- Exposición de las demostraciones de los resultados dados.
- Grafos Hamiltonianos.
- Coloraciones y emparejamientos.
- Platitud.

¿ALGUNA PREGUNTA?