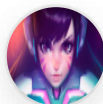


WUOLAH



Belen_Dominguez

www.wuolah.com/student/Belen_Dominguez



3916

Tema 3 - Apuntes.pdf

MD Apuntes Completos Temario



2º Matemática Discreta



Grado en Ingeniería Informática - Ingeniería del Software



Escuela Técnica Superior de Ingeniería Informática
US - Universidad de Sevilla

 escuela
de negocios
CÁMARA DE SEVILLA

MÁSTER EN DIRECCIÓN Y GESTIÓN DE RECURSOS HUMANOS

www.mastersevilla.com

Inscríbete

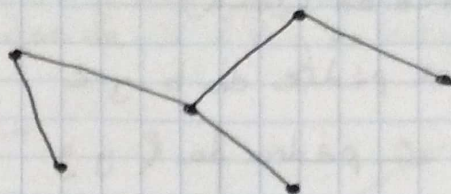


BECAS

TEMA 3 - Árboles

• Introducción

Un grafo $T=(V,A)$ es un árbol si $\begin{cases} \rightarrow \text{Es conexo} \\ \rightarrow \text{No tiene ciclos} \end{cases}$



Un grafo $G=(V,A)$ se llamará bosque si no contiene ciclos. Cada componente conexa del bosque es un árbol.

* Teorema de caracterización de árboles

a) T es un árbol (conexo y sin ciclos).

b) Entre dos vértices cualesquiera existe un único camino simple.

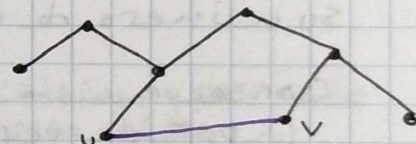
c) Al eliminar una arista el grafo se desconecta y da lugar a dos componentes conexas (2 árboles).

d) $n_{\text{aristas}} = n_{\text{vértices}} - 1$ y T es conexo.

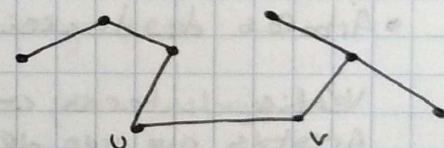
e) $n_{\text{aristas}} = n_{\text{vértices}} - 1$ y T no contiene ciclos.

* Algunas propiedades

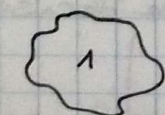
- Si a un árbol se añade una arista se genera un ciclo.



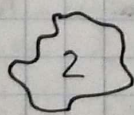
- Si eliminamos otra arista del ciclo que creemos se obtiene otro árbol (no tiene por que ser isomorfo).



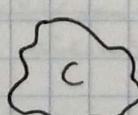
- Si $G=(V,A)$ es un bosque con c componentes conexas (árboles), entonces $n_v = n_a + c$.



$$v_1 = d_1 + 1$$



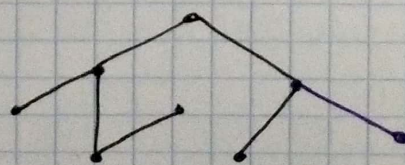
$$v_2 = d_2 + 2$$



$$v_c = d_c + 1$$

$$\begin{aligned} n_v &= v_1 + v_2 + \dots + v_c = \\ &= d_1 + 1 + d_2 + 1 + \dots + d_c + 1 = \\ &= n_a + c \end{aligned}$$

- Si añadimos una arista seguida de un vértice, obtenemos un nuevo árbol.



$$n_v = n_a + 1$$

$$n'_v = n'_a + 1$$

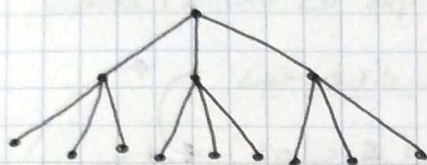
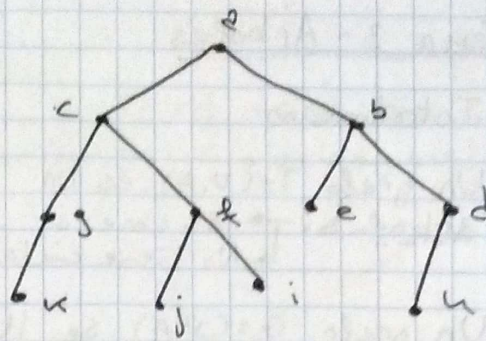
• Árboles enraizados

Árbol que posee un vértice destacado (raíz)

a es el padre de b y c

c es el padre de f y g

etc.



Árbol m-ario \rightarrow Los vértices tienen todos m hijos

$m = 3$: árbol ternario

Distinguimos entre árbol m-ario completo (ejemplo de arriba) y árbol m-ario incompleto



Algunas propiedades de los árboles m-ario:

- * Sea $T = (V, A)$ un árbol m-ario, de altura p y h hojas se verifican las siguientes dos desigualdades:

$$p \geq \log_m h \quad h \leq m^p$$

- * Si tenemos otro árbol m-ario con i vértices internos, su número de vértices viene dado por: $n = mi + 1$

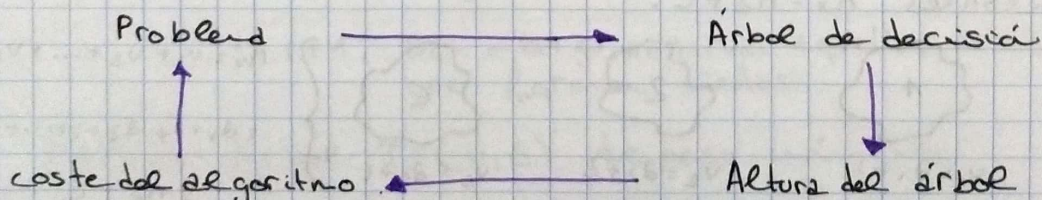
Consecuencia \rightarrow Un árbol binario tiene un número impar de vértices ($2i + 1$, i internos e $i + 1$ hojas)

• Árboles de decisión

Vértice interno \leftrightarrow una decisión

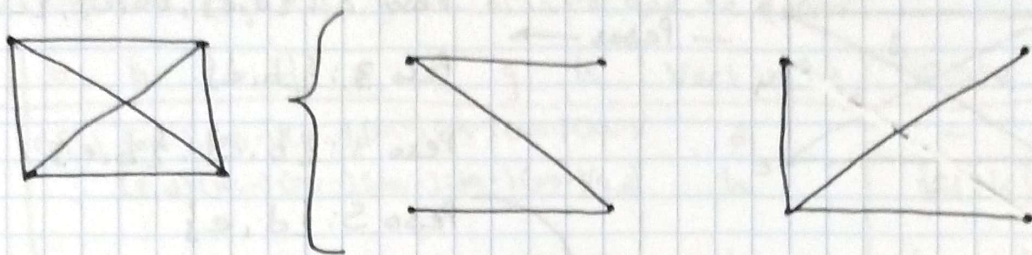
Aristas que van al siguiente nivel \leftrightarrow resultados de las decisiones

Hojas \leftrightarrow Resultado final del proceso



- Árboles recubridores

Llamamos árbol recubridor a un grafo $G' = (V', A')$ que es un subgrafo de $G = (V, A)$ donde $V' = V$ y no hay ciclos.



Teorema: Un grafo $G = (V, A)$ no contendrá algún árbol recubridor si, y sólo si, es conexo.

Si el grafo no es conexo, contiene un algún bosque recubridor.

Los algoritmos DFS y BFS vistos anteriormente se pueden usar para buscar árboles recubridores.

- Grafos ponderados

- Árbol recubridor de peso mínimo

Algoritmo de Kruskal:

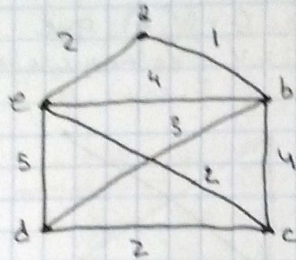
Entrada $\rightarrow G = (V, A)$ grafo conexo con n vértices y para cada arista $a \in A$ lleva asociado un peso $p(a)$.

Pasos \rightarrow Ordenamos las aristas de menor a mayor peso

- $\rightarrow i = 0, E \leftarrow \emptyset, T(V, E) \rightarrow$ Vamos cada peso, dibujando a parte un nuevo grafo
- \rightarrow Mientras haya vértices por recorrer,
 - $i = i + 1$; Sin embargo, no sirven aquellas aristas que forman ciclos

Salida: Árbol $T = (V, E)$ de peso mínimo.

Ejemplo: Dado el siguiente grafo, hallar el árbol de peso mínimo



— Pesos —>

Peso 1: $\{a, b\}$

Peso 2: $\{a, e\}, \{e, c\}, \{c, d\}$

Peso 3: $\{b, d\}$

Peso 4: $\{b, c\}, \{b, e\}$

Peso 5: $\{d, e\}$

Dibujamos los vértices y vamos viendo las aristas que no forman ciclos hasta que se hallan recorrido todos los vértices

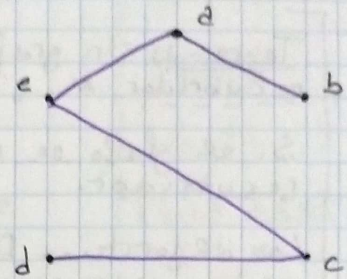
$\{a, b\}$

$\{a, e\}, \{e, c\}, \{c, d\}$

$\{b, d\}$

$\{b, c\}, \{b, e\}$

$\{d, e\}$



Peso total = $1 \cdot 1 + 2 \cdot 3 = 7$

• Camino más corto

Sea $G = (V, A)$ un grafo con n vértices, y para cada arista $a \in A$ tiene asociado un peso $p(a)$ (también llamado longitud).

Longitud de camino (l(c)): Suma de los pesos de las aristas que componen el camino de x a y .

$$C: a - b - c \rightarrow l(c) = 1 + 4 = 5$$

Distancia entre dos vértices ($d(x, y)$): menor de las longitudes de los caminos entre x y y .

$$d(a, e) \rightarrow \text{Camino mínimo: } a - e - c$$

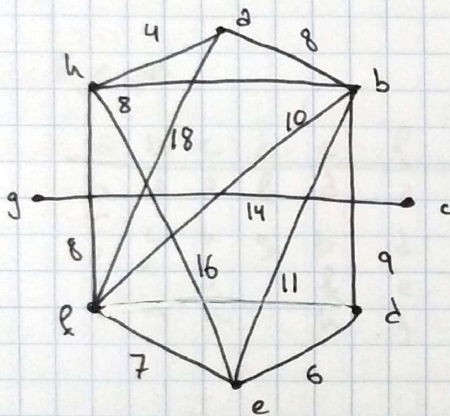
Si x y y no están en la misma componente conexa:

$$d(x, y) = \infty$$

Algoritmo de Dijkstra con un vértice fijo v_0

Utilizaremos un ejemplo para explicarlo. Primero crearemos una tabla donde tendremos un contador (i), la lista de vértices, el vértice fijo (u) y la arista que se forma:

i	a	b	c	d	e	f	g	h	Vert u ES	Arista
0	(0,-)	(∞,-)	(∞,-)	(∞,-)	(∞,-)	(∞,-)	(∞,-)	(∞,-)	a	—
1		(8,d)	(∞,-)	(∞,-)	(∞,-)	(∞,-)	(∞,-)	(4,d)	h	{a,h}
2		(8,d)	(∞,-)	(∞,-)	(20,h)	(12,h)	(∞,-)		b	{a,b}
3			(∞,-)	(17,b)	(19,b)	(12,h)	(∞,-)		h	{h,f}
4			(∞,-)	(17,b)	(19,b)		(∞,-)		d	{b,d}
5			(∞,-)		(19,b), f	(∞,-)			e	{b,e} {f,e}



El contador irá aumentado hasta llegar al número de vértices - 1 como mucho.

- 1- Si queremos ver el camino más corto de a hacia e, empezamos con $u = a$. En el primer paso siempre pondremos la distancia a los demás vértices como infinito.
- 2- Veremos los primeros vértices adyacentes a 'a'. Indicaremos la distancia que llevamos recorrida y el vértice del que salimos. Nos quedaremos con el camino más corto.

3- Ahora tendremos que comparar si el camino que establecemos en el paso anterior es más pequeño que el que se forma con el nuevo vértice (h). En este caso, el camino más corto de vértice b es el que tenemos antes ({a,b}) que el que se formaría {a,h,b}. Además también es el camino más corto de todos.

4- Volvemos a repetir el paso anterior.

5- En esta ocasión, nos sale el mismo peso desde los dos caminos.

6- Aunque salió como camino más corto {b,d}, hacer el camino {a-b-d-e} = 23 es más largo que los dos caminos de longitud 19 de los recorridos {b,e} y {f,e}.

Por tanto tenemos los siguientes pesos:

$$d(a, h) = 4 \quad d(a, f) = 12 \quad d(a, e) = 19 \quad d(a, b) = 8 \quad d(a, d) = 17$$

Nos salen los siguientes caminos 7

