

REDES NO VALORADAS. REDES VALORADAS. ALGORITMOS.



Asignatura Grupo
Apellidos Nombre
Ejercicio del día

REDES NO VALORADAS

- Definiciones y teoremas
- Diagramas. Matemática discreta.
- Tipo de redes
- Conceptos no orientados
- Algoritmos
- Aplicaciones

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

GRAFO - TAD GRAFO

Un Grafo es una entidad matemática introducida por Euler en 1736 para representar entidades (vértices) que pueden relacionarse libremente entre sí, mediante el concepto de arista.

Se puede definir un TAD (Tipo Abstracto de Datos) Grafo basado en estas ideas, el cual contiene elementos sobre los que esta definida una relación de vecindad o adyacencia. Un vértice puede relacionarse con cualquier otro vértice y establecer cualquier número de relaciones.

Hay muchas situaciones en las cuales el modelado más conveniente de los datos de una aplicación es mediante grafos, por ejemplo la representación de una red de carreteras, calles, telecomunicaciones, electrificación, internet, planificación de tareas, etapas de un proceso industrial, etc.

GRAFO - MATRIZ DE ADYACENCIA

Un Grafo es un conjunto de objetos, llamados vértices o nodos, conectados entre sí.

A las conexiones entre los objetos se las denomina *aristas o arcos*.

Si en un Grafo se indica con un 1 la existencia de conexión y con un 0 la falta de existencia de ella, este se puede representar con una matriz que proporciona la misma información que el Grafo.

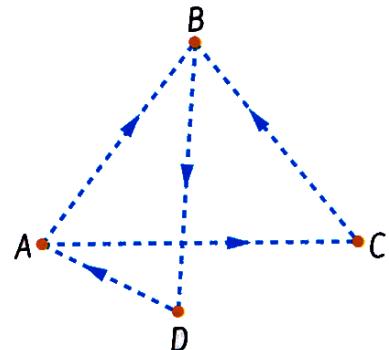
Ejemplo: Un archipiélago formado por cuatro islas A, B, C y D establece un sistema de comunicación por avión. Por falta de medios, no todas las islas están conectadas entre sí. El grafo que aparece al margen representa las conexiones que se ofrecen.

Las islas A, B, C y D son los vértices del grafo y las conexiones

$A \rightarrow B$, $A \rightarrow C$, $B \rightarrow D$, $C \rightarrow B$, $D \rightarrow A$

son *las aristas o arcos del grafo*.

La matriz de adyacencia es:



	A	B	C	D	← Destino
Origen →	A	0	1	1	0
	B	0	0	0	1
	C	0	1	0	0
	D	1	0	0	0

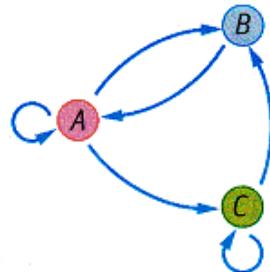
$a_{12} = 1$ indica que existe comunicación de la isla A hacia B
 $a_{21} = 0$ indica que no existe comunicación de la isla B hacia A

Sí las aristas o arcos de un grafo están orientadas, como en el ejemplo, el grafo se denomina Grafo Dirigido o Dígrafo (sus aristas vienen dados por pares ordenados).

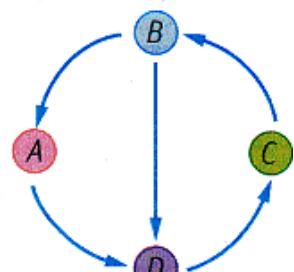
MATRIZ DE ADYACENCIA ASOCIADA A LOS GRAFOS

Los Grafos pueden ser no dirigidos o dirigidos

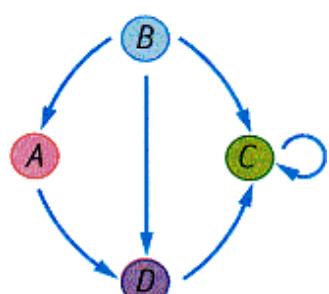
Grafo dirigido o Dígrafo: Las aristas o arcos están orientadas.



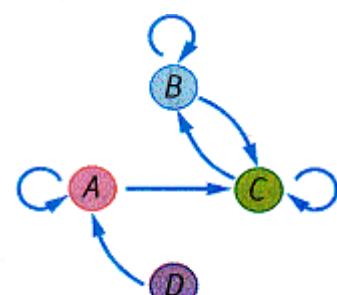
$$M = \text{Origen} \rightarrow \begin{array}{c} \begin{matrix} & A & B & C & \leftarrow \text{Destino} \\ \begin{matrix} A \\ B \\ C \end{matrix} & \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \end{pmatrix} \end{matrix} \end{array}$$



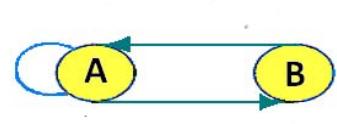
$$M = \text{Origen} \rightarrow \begin{array}{c} \begin{matrix} & A & B & C & D & \leftarrow \text{Destino} \\ \begin{matrix} A \\ B \\ C \\ D \end{matrix} & \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{matrix} \end{array}$$



$$M = \text{Origen} \rightarrow \begin{array}{c} \begin{matrix} & A & B & C & D & \leftarrow \text{Destino} \\ \begin{matrix} A \\ B \\ C \\ D \end{matrix} & \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{matrix} \end{array}$$

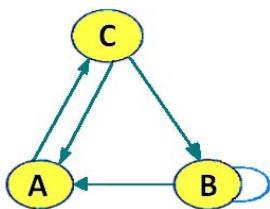


$$M = \text{Origen} \rightarrow \begin{array}{c} \begin{matrix} & A & B & C & D & \leftarrow \text{Destino} \\ \begin{matrix} A \\ B \\ C \\ D \end{matrix} & \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \end{matrix} \end{array}$$

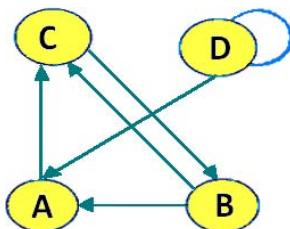


$$M = \begin{array}{c} \begin{matrix} & A & B \\ \begin{matrix} A \\ B \end{matrix} & \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \end{matrix} \end{array}$$

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día



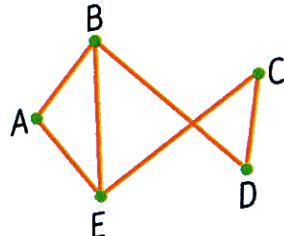
$$M = \begin{array}{c} \begin{matrix} & A & B & C \\ A & 0 & 0 & 1 \\ B & 1 & 1 & 0 \\ C & 1 & 1 & 0 \end{matrix} \end{array}$$



$$M = \begin{array}{c} \begin{matrix} & A & B & C & D \\ A & 0 & 0 & 1 & 0 \\ B & 1 & 0 & 1 & 0 \\ C & 0 & 1 & 0 & 0 \\ D & 1 & 0 & 0 & 1 \end{matrix} \end{array}$$

GRAFO NO DIRIGIDO

Pueblos unidos por carreteras



$$\begin{array}{c} \begin{matrix} & A & B & C & D & E \\ \text{Origen} \rightarrow & 0 & 1 & 0 & 0 & 1 \\ & 1 & 0 & 0 & 1 & 1 \\ & 0 & 0 & 0 & 1 & 1 \\ & 0 & 1 & 1 & 0 & 0 \\ & 1 & 1 & 1 & 0 & 0 \end{matrix} \leftarrow \text{Destino} \end{array}$$

Un grafo $G(V, A)$ está formado por un conjunto no vacío V de vértices o nodos y otro conjunto A de aristas o lados constituidos por pares no ordenados de vértices distintos.

Algebraicamente, un grafo es la terna $G = (V, A, \varphi)$, donde V es un conjunto no vacío de vértices, A es un conjunto de aristas y φ es la función de incidencia.

$$\varphi: A \longrightarrow V^* \quad \left\{ \begin{array}{l} V^* \text{ es un conjunto formado por} \\ \text{subconjuntos de 1 o 2 elementos de } V \end{array} \right.$$

Los Grafos se pueden representar gráficamente mediante diagramas y en forma matricial con la matriz de adyacencia y la matriz de incidencia.

MATRIZ DE ADYACENCIA

Sea un grafo $G = (V, A, \varphi)$ con un conjunto de vértices $V = \{v_1, v_2, v_3, \dots, v_n\}$ y un conjunto de aristas $A = \{a_1, a_2, a_3, \dots, a_m\}$, se define la matriz de adyacencia del grafo G como una matriz booleana de $n \times n$:

$$M_a(G) = (m_{ij}) \text{ donde } m_{ij} = \begin{cases} 1 & \text{si } v_i \text{ es adyacente a } v_j \\ 0 & \text{si } v_i \text{ no es adyacente a } v_j \end{cases}$$

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

GRAFO SIMPLE: Grafo sin aristas paralelas y sin bucles.

Todas las matrices de adyacencia M de un GRAFO SIMPLE (sin aristas paralelas y sin bucles) son matrices simétricas con diagonal principal formada por ceros.

El resto de elementos son unos, formando dos triángulos (uno superior y otro inferior).

$$\text{GRAFO SIMPLE : } M_a = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix} \rightarrow \text{Tr}(M_a) = 0 + 0 + 0 + 0 = 0$$

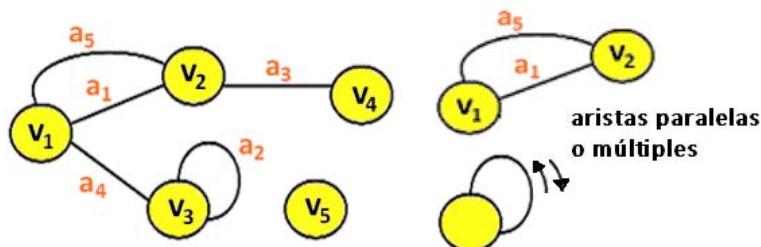
Para determinar si un grafo tiene o no tiene triángulos, los ALGORITMOS (secuencia de pasos lógicos que permiten solucionar un problema) a menudo devuelven como salida tres vértices que cumplen dicha condición.

Otro enfoque es encontrar la traza de M^3 , el Grafo no contiene triángulos $\Leftrightarrow \text{Tr}(M^3) = 0$

$$M_a^3 = \begin{pmatrix} m_{11} & & & \\ & m_{22} & & \\ & & m_{33} & \\ & & & m_{44} \end{pmatrix} \rightarrow \text{Tr}(M_a^3) = m_{11} + m_{22} + m_{33} + m_{44}$$

GRAFO NO DIRIGIDO

Matriz de adyacencia del grafo
(relaciona vértices con vértices)



$$M_a(G) = \begin{matrix} \begin{array}{cccccc} v_1 & v_2 & v_3 & v_4 & v_5 \\ \hline v_1 & 0 & 2 & 1 & 0 & 0 \\ v_2 & 2 & 0 & 0 & 1 & 0 \\ v_3 & 1 & 0 & 2 & 0 & 0 \\ v_4 & 0 & 1 & 0 & 0 & 0 \\ v_5 & 0 & 0 & 0 & 0 & 0 \end{array} \end{matrix} \rightarrow \begin{array}{l} \text{gr}(1)=3 \\ \text{gr}(2)=3 \\ \text{gr}(3)=3 \\ \text{gr}(4)=1 \\ \text{gr}(5)=0 \end{array} \quad \sum_{i=1}^5 \text{gr}(v_i) = 10 \quad \sum_{i=1}^5 \text{gr}(v_i) = 2.5 = 10$$

5 aristas

Cada arista hace incrementar en uno el grado de cada vértice de sus extremos (si son distintos) o en 2 si se trata de un bucle.

En un Grafo $\begin{cases} \text{Un vértice es aislado} \Leftrightarrow \text{grado(vértice)} = 0 \\ \text{Un vértice es colgante o pendiente} \Leftrightarrow \text{grado(vértice)} = 1 \end{cases}$

El Grado (gr) de un vértice es la cantidad de aristas incidentes en el vértice.

En todo Grafo la suma de los grados de los vértices es igual al doble de la cantidad de aristas, es decir,
 $\sum \text{gr}(v_i) = 2 |A|$

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

Bucle o Lazo: Arista que está asociada a dos vértices idénticos (a_2 es un bucle).

Vértices Adyacentes: Son los vértices unidos por alguna arista

v_2 es adyacente a v_1 y v_4 pero no es adyacente a v_3

Aristas Adyacentes: Son las aristas que tienen un único vértice en común, siendo distintas y no paralelas.

a_1 y a_3 son aristas adyacentes

Aristas Incidentes: Son las aristas que tienen un vértice como extremo.

v_2 es adyacente a v_1 y v_4 pero no es adyacente a v_3

Aristas Múltiples o Paralelas: Dos o más aristas que son incidentes en al menos dos vértices. Las aristas paralelas tienen los mismos vértices en común o inciden sobre los mismos vértices. Son muy utilizadas en redes eléctricas.

a_i paralela a $a_j \Leftrightarrow \phi(a_i) = \phi(a_j)$ con $a_i \neq a_j$

a_1 y a_5 son aristas paralelas o múltiples.

Grafo Sencillo o Simple: No tiene aristas múltiples, tiene una sola arista entre dos vértices. En general, un grafo que no tiene bucles ni aristas paralelas o dirigidas.

Multigrafo o Pseudografo: Grafo que contiene aristas paralelas. De esta forma, dos nodos pueden estar conectados por más de una arista.

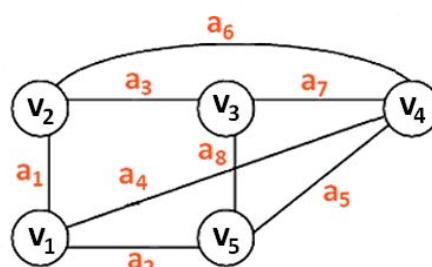
Los Multigrafos podrían utilizarse, por ejemplo, para modelar las posibles conexiones de vuelo ofrecidas por una aerolínea. En este caso se tendría un grafo dirigido, donde cada nodo es una localidad y donde pares de aristas paralelas conectan localidades, según un vuelo es hacia o desde una localidad a la otra.

MATRIZ DE INCIDENCIA

Sea un grafo $G = (V, A, \phi)$ con un conjunto de vértices $V = \{v_1, v_2, v_3, \dots, v_n\}$ y un conjunto de aristas $A = \{a_1, a_2, a_3, \dots, a_m\}$, se define la matriz de incidencia del grafo G como una matriz booleana de $n \times n$:

$$M_i(G) = (m_{ij}) \text{ donde } m_{ij} = \begin{cases} 1 & \text{si } v_i \text{ es incidente a } a_j \\ 0 & \text{si } v_i \text{ no es incidente a } a_j \end{cases}$$

Matriz de adyacencia y matriz de incidencia
del grafo no dirigido



Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

- Matriz de Adyacencia (relaciona vértices con vértices):

$$M_a(G) = \begin{matrix} & \begin{matrix} v_1 & v_2 & v_3 & v_4 & v_5 \end{matrix} \\ \begin{matrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{matrix} & \left(\begin{matrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \end{matrix} \right) \end{matrix} \rightarrow \begin{matrix} gr(1)=3 \\ gr(2)=3 \\ gr(3)=3 \\ gr(4)=4 \\ gr(5)=3 \end{matrix}$$

$\sum_{i=1}^5 gr(v_i) = 2.8 = 16$
8 aristas

- Matriz de Incidencia (relaciona vértices con aristas):

$$M_i(G) = \begin{matrix} & \begin{matrix} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & a_7 & a_8 \end{matrix} \\ \begin{matrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{matrix} & \left(\begin{matrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{matrix} \right) \end{matrix}$$

CAMINOS DE UN GRAFO

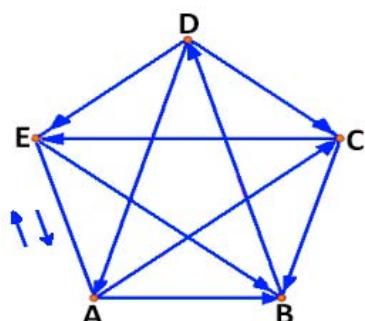
Las sucesivas potencias de la Matriz de Adyacencia (M) de un Grafo dirigido (Dígrafo) indican el número de caminos diferentes que se pueden seguir para ir de un vértice a otro. En esta línea,

M^2 indica el número de caminos diferentes de longitud 2 para ir de un vértice a otro. Es decir, el número de caminos diferentes para ir de un vértice a otro pasando por otro vértice.

M^3 indica el número de caminos diferentes de longitud 3 para ir de un vértice a otro. Es decir, el número de caminos diferentes para ir de un vértice a otro pasando, en su intermedio, por dos vértices.

Ejemplo: El Jefe de Recursos Humanos de una empresa quiere saber quién es el líder de un grupo de 5 personas (A, B, C, D, E). Para ello, hace entrevistas personales por parejas, estableciendo quién tiene ascendencia sobre el otro. Representa las encuestas en un grafo dirigido de cinco vértices, donde $A \rightarrow B$ indica que A domina a B.

¿Quién es el Líder?



Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

Matriz de Adyacencia: $M = \begin{array}{c} \begin{matrix} & \text{A} & \text{B} & \text{C} & \text{D} & \text{E} \end{matrix} \\ \begin{matrix} \text{A} & \left(\begin{matrix} 0 & 1 & 1 & 0 & 1 \end{matrix} \right) \\ \text{B} & \left(\begin{matrix} 0 & 0 & 0 & 1 & 0 \end{matrix} \right) \\ \text{C} & \left(\begin{matrix} 0 & 1 & 0 & 0 & 1 \end{matrix} \right) \\ \text{D} & \left(\begin{matrix} 1 & 0 & 1 & 0 & 1 \end{matrix} \right) \\ \text{E} & \left(\begin{matrix} 1 & 1 & 0 & 0 & 0 \end{matrix} \right) \end{array} \rightarrow \begin{array}{l} \text{A domina a 3 personas} \\ \text{B domina a 1 persona} \\ \text{C domina a 2 personas} \\ \text{D domina a 3 personas} \\ \text{E domina a 2 personas} \end{array}$

Para analizar quien es el líder del grupo se suman las líneas de la Matriz de Adyacencia, obteniendo cuantos trabajadores son influenciados por una persona (domina). Tanto A como D dominan a 3 personas, no pudiendo decidir quién es en líder. Se hace necesario calcular M^2

$$M^2 = M \times M = \begin{pmatrix} 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 3 & 1 & 0 & 2 \\ 0 & 1 & 1 & 1 & 1 \end{pmatrix} \rightarrow \begin{array}{l} 5 \\ 3 \\ 3 \\ 7 \\ 4 \end{array}$$

Los elementos de M^2 indican el número de caminos diferentes de longitud 2 que se pueden seguir para ir de un vértice a otro. Es decir, dominios de segundo orden.

Más en concreto, cantidad de trabajadores que son influenciados por personas que domina cierta persona. Por ejemplo, si A domina a B y B domina a C, es lógico pensar que A domina a C.

$$M + M^2 = \begin{array}{c} \begin{matrix} & \text{A} & \text{B} & \text{C} & \text{D} & \text{E} \end{matrix} \\ \begin{matrix} \text{A} & \left(\begin{matrix} 0 & 1 & 1 & 0 & 1 \end{matrix} \right) \\ \text{B} & \left(\begin{matrix} 0 & 0 & 0 & 1 & 0 \end{matrix} \right) \\ \text{C} & \left(\begin{matrix} 0 & 1 & 0 & 0 & 1 \end{matrix} \right) \\ \text{D} & \left(\begin{matrix} 1 & 0 & 1 & 0 & 1 \end{matrix} \right) \\ \text{E} & \left(\begin{matrix} 1 & 1 & 0 & 0 & 0 \end{matrix} \right) \end{array} + \begin{pmatrix} 1 & 2 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 3 & 1 & 0 & 2 \\ 0 & 1 & 1 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 3 & 1 & 1 & 2 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 2 & 0 & 1 & 1 \\ 2 & 3 & 2 & 0 & 3 \\ 1 & 2 & 1 & 1 & 1 \end{pmatrix} \rightarrow \begin{array}{l} 8 \\ 4 \\ 5 \\ 10 \\ 6 \end{array}$$

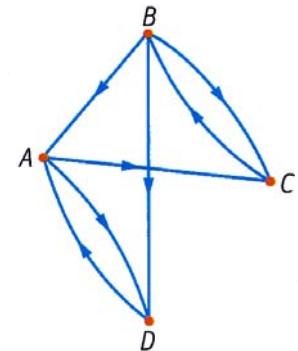
La matriz ($M + M^2$) representa la cantidad de dominios de primer y segundo orden que ejerce cada trabajador sobre el resto de personas del grupo.

Pudiendo afirmar que el trabajador D es el líder del grupo.

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día



- a) Indica todos los caminos de longitud 3 que se pueden seguir de C a D.
 b) Indica todos los caminos de longitud 4 que se pueden seguir de C a A.



Matriz de Adyacencia: $M = \begin{matrix} \text{Origen} & \rightarrow \end{matrix}$

$$\begin{array}{c} \text{A} \quad \text{B} \quad \text{C} \quad \text{D} \quad \leftarrow \text{Destino} \\ \begin{matrix} \text{A} \\ \text{B} \\ \text{C} \\ \text{D} \end{matrix} \left(\begin{array}{cccc} 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{array} \right) \end{array}$$

a) Para encontrar los caminos de longitud 3 hay que hacer M^3

$$M^2 = M \times M = \left(\begin{array}{cccc} 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{array} \right) \times \left(\begin{array}{cccc} 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{array} \right) = \left(\begin{array}{ccccc} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{array} \right)$$

$$M^3 = M^2 \times M = \left(\begin{array}{cccc} 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{array} \right) \times \left(\begin{array}{cccc} 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{array} \right) = \begin{array}{c} \text{A} \quad \text{B} \quad \text{C} \quad \text{D} \quad \leftarrow \text{Destino} \\ \begin{matrix} \text{A} \\ \text{B} \\ \text{C} \\ \text{D} \end{matrix} \left(\begin{array}{ccccc} 1 & 0 & 2 & 2 & 2 \\ 2 & 1 & 2 & 2 & 2 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \end{array} \right) \end{array}$$

El número de caminos de longitud 3 que se pueden seguir para ir de C a D viene dado por elemento de la tercera fila y cuarta columna de M^3 , es decir, $a_{34} = 1$.

Hay un único camino $C \rightarrow B \rightarrow A \rightarrow D$

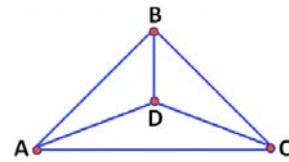
b) Para encontrar los caminos de longitud 4 hay que hacer M^4

$$M^4 = M^3 \times M = \left(\begin{array}{cccc} 1 & 0 & 2 & 2 \\ 2 & 1 & 2 & 2 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \end{array} \right) \times \left(\begin{array}{cccc} 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{array} \right) = \begin{array}{c} \text{A} \quad \text{B} \quad \text{C} \quad \text{D} \quad \leftarrow \text{Destino} \\ \begin{matrix} \text{A} \\ \text{B} \\ \text{C} \\ \text{D} \end{matrix} \left(\begin{array}{ccccc} 1 & 2 & 1 & 1 & 1 \\ 3 & 2 & 3 & 3 & 3 \\ 2 & 1 & 2 & 2 & 2 \\ 1 & 0 & 2 & 2 & 2 \end{array} \right) \end{array}$$

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

El número de caminos de longitud 4 que se pueden seguir de C a A viene dado por el elemento de la tercera fila y primera columna de M^4 , es decir, $a_{31} = 2$.

Con lo que hay dos caminos: $\begin{cases} C \rightarrow B \rightarrow A \rightarrow D \rightarrow A \\ C \rightarrow B \rightarrow C \rightarrow B \rightarrow A \end{cases}$



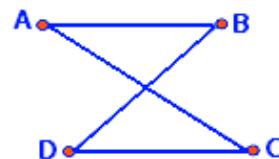
- Escribir la matriz de adyacencia del grafo adjunto y decir cómo es esa matriz.

$$\text{Matriz de Adyacencia : } M(G) = \begin{array}{cc|cccc} & & A & B & C & D \\ \hline A & 0 & 1 & 1 & 1 \\ B & 1 & 0 & 1 & 1 \\ C & 1 & 1 & 0 & 1 \\ D & 1 & 1 & 1 & 0 \end{array} \rightarrow \text{Tr}(M) = 0 + 0 + 0 + 0 = 0$$

Como ocurre con todas las Matrices de Adyacencia de un Grafo Simple (sin aristas paralelas y sin bucles), se trata de una matriz simétrica con diagonal principal formada por ceros. El resto de elementos son unos, formando dos triángulos (uno superior y otro inferior).

$$\text{Matriz de Adyacencia en forma simplificada: } M(G) = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix}$$

- Escribir la matriz de adyacencia del grafo adjunto y deducir los campos de longitud 4 de A a D.
 ¿Posee triángulos el grafo?



$$\text{Matriz de Adyacencia : } M_a(G) = \begin{array}{cc|cccc} & & A & B & C & D \\ \hline A & 0 & 1 & 1 & 0 \\ B & 1 & 0 & 0 & 1 \\ C & 1 & 0 & 0 & 1 \\ D & 0 & 1 & 1 & 0 \end{array}$$

El grafo no contiene triángulos $\Leftrightarrow \text{Tr}[M_a^*(G)] = 0$

$$M_a^2(G) = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 2 & 0 & 0 & 2 \\ 0 & 2 & 2 & 0 \\ 0 & 2 & 2 & 0 \\ 2 & 0 & 0 & 2 \end{pmatrix}$$

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

$$M_a^3(G) = \begin{pmatrix} 2 & 0 & 0 & 2 \\ 0 & 2 & 2 & 0 \\ 0 & 2 & 2 & 0 \\ 2 & 0 & 0 & 2 \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 4 & 4 & 0 \\ 4 & 0 & 0 & 4 \\ 4 & 0 & 0 & 4 \\ 0 & 4 & 4 & 0 \end{pmatrix}$$

El grafo no contiene triángulos $\Leftrightarrow \text{Tr}[M_a^3(G)] = 0 + 0 + 0 + 0 = 0$

Para calcular los caminos de longitud 4 es necesario calcular M_a^4 :

$$M_a^4(G) = \begin{pmatrix} 0 & 4 & 4 & 0 \\ 4 & 0 & 0 & 4 \\ 4 & 0 & 0 & 4 \\ 0 & 4 & 4 & 0 \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix} = \begin{matrix} \text{A} & \begin{pmatrix} 8 & 0 & 0 & \boxed{8} \end{pmatrix} \\ \text{B} & \begin{pmatrix} 0 & 8 & 8 & 0 \end{pmatrix} \\ \text{C} & \begin{pmatrix} 0 & 8 & 8 & 0 \end{pmatrix} \\ \text{D} & \begin{pmatrix} 8 & 0 & 0 & 8 \end{pmatrix} \end{matrix}$$

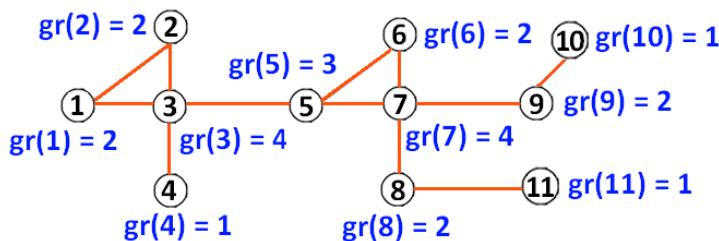
Caminos de longitud 4 de A a D: $m_{14} = 8 \begin{cases} \text{ABABD} & \text{ABDBD} & \text{ACABD} & \text{ACDBD} \\ \text{ABACD} & \text{ABDCD} & \text{ACACD} & \text{ACDCD} \end{cases}$

- Un grafo tiene 12 aristas, 2 vértices de grado 4, 1 vértice de grado 3, 5 vértices de grado 2 y el resto de vértices son colgantes. ¿Cuál es la cantidad total de vértices del grafo?. Dibujar una posibilidad.

$$\sum \text{gr}(v_i) = 2|A|$$

$$2|A| = (2 \times 4) + (1 \times 3) + (5 \times 2) + (b \times 1) = 2 \times 12 \rightarrow b = 3 \text{ Vértices colgantes (grado 1)}$$

Número total vértices: $\sum v_i = 2 + 1 + 5 + 3 = 11$ Vértices.



CICLO o CIRCUITO. REGIÓN

Los Grafos pueden ser no dirigidos o dirigidos. Los Grafos dirigidos reciben el nombre de Dígrafos. Para muchas aplicaciones es necesario indicar el sentido de las aristas.

En los Dígrafos o Grafos dirigidos se indica el sentido de las aristas.

Si el Grafo indica, por ejemplo, las calles de un pueblo y los vértices las esquinas, para obtener el camino más corto en automóvil entre dos esquinas dadas es necesario conocer el sentido de las calles.

Un Dígrafo es la terna $G = (V, A, \delta)$, donde V es un conjunto no vacío de vértices, A es un conjunto de aristas y δ la función de incidencia: $\delta : A \longrightarrow V \times V$

La función de incidencia se dice dirigida, haciendo corresponder a cada arista un *Par Ordenado* de vértices, el primero se llama *Extremo Inicial* de la arista y el segundo *Vértice Final*.

Los Caminos y Ciclos se definen de la misma forma que para los Grafos no dirigidos, respetando el sentido de las aristas.

CAMINO: Sucesión de aristas adyacentes distintas.

CICLO o CIRCUITO: Es un camino cerrado, el vértice inicial coincide con el final.

LONGITUD DEL CAMINO: Cantidad de aristas que lo componen.

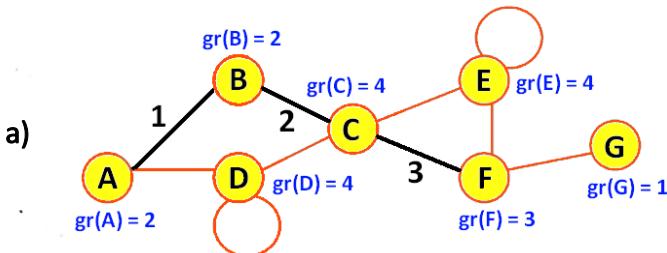
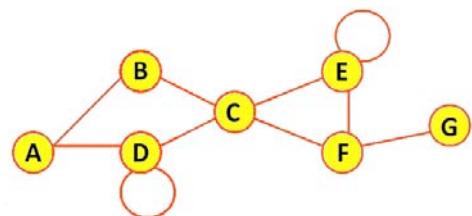
CAMINO SIMPLE: Cuando todos los vértices son distintos.

CAMINO ELEMENTAL: Cuando todas las aristas son distintas.

GRADO DE UNA REGIÓN: Longitud del camino que la bordea.

a) Grado de cada vértice y caminos de A hacia F de longitud 3 y 5

b) Ciclos de longitud 3, 4, 5 y 7

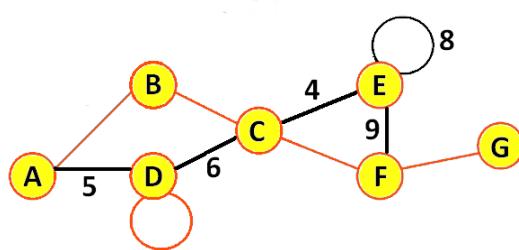


Un Camino posible es $C = (A; B; C; F)$, también se puede nombrar un camino mediante vértices y aristas: $C = (A, 1 ; B, 2 ; C, 3 ; F)$

$\text{Long}(C) = 3$ porque tiene 3 aristas.

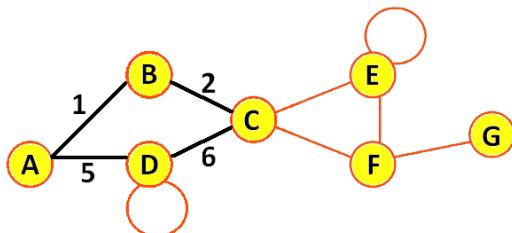
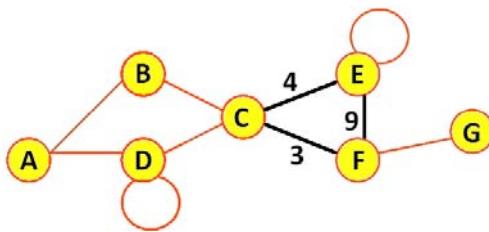
Otro Camino posible de longitud 5 puede ser

$C = (A, 5 ; D, 6 ; C, 4 ; E, 8 ; E, 9 ; F)$



b) Un posible Ciclo o Circuito de longitud 3 sería:

$$C = (C, 4 ; E, 9 ; F, 3 ; C)$$

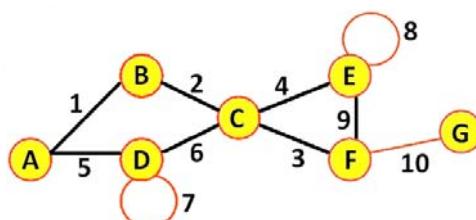
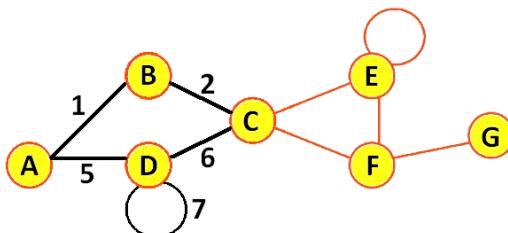


Un posible Ciclo o Circuito de longitud 4 sería

$$C = (A, 5 ; B, 2 ; C, 6 ; D, 5 ; A)$$

Un posible Ciclo o Circuito de longitud 5 sería:

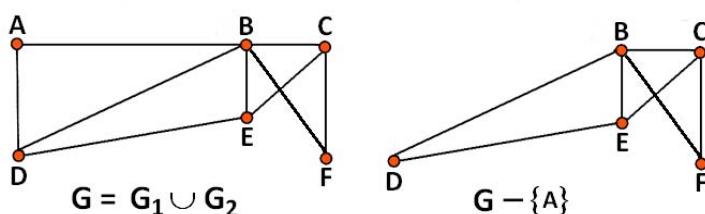
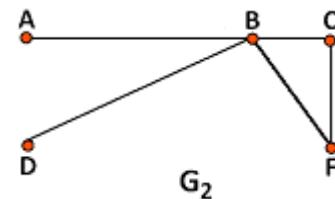
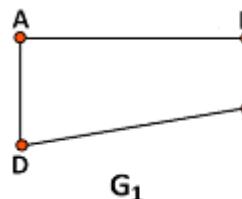
$$C = (A, 1 ; B, 2 ; C, 6 ; D, 7 ; D, 5 ; A)$$



Un posible ciclo o circuito de longitud 7 sería:

$$C = (A, 1 ; B, 2 ; C, 4 ; E, 9 ; F, 3 ; C, 6 ; D, 5 ; A)$$

- Dados los grafos G_1 y G_2 adjuntos, encontrar el grafo unión $G = G_1 \cup G_2$ y el grafo $G - \{A\}$



Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

GRAFO k - REGULAR

Un grafo $G = (V, A, \phi)$ es k -regular $\Leftrightarrow \text{gr}(v_i) = k \quad \forall v_i \in V \quad \forall k \in \mathbb{N}$

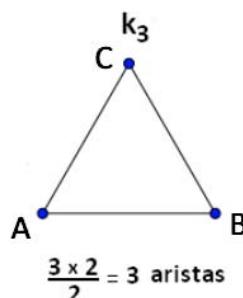
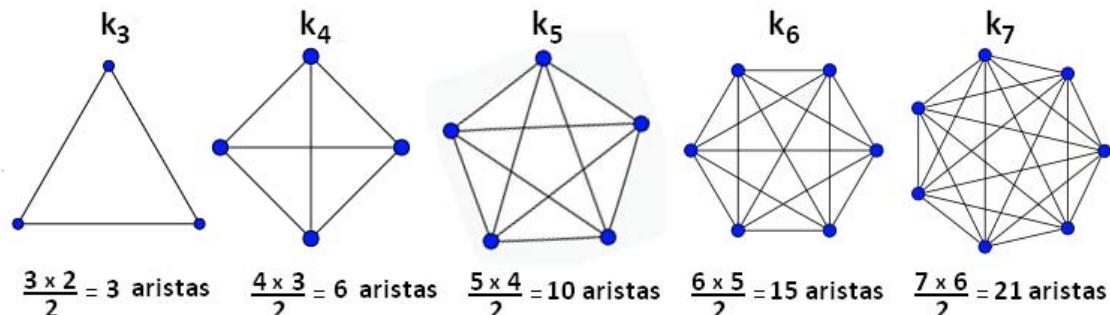
GRAFO COMPLETO

Es un grafo simple donde cada par de vértices está conectado por una arista. Es un grafo k -regular con todos sus vértices de grado $(n-1)$

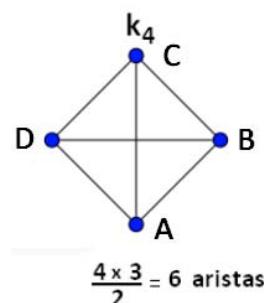
Un grafo completo de n vértices tiene $\frac{n(n-1)}{2}$ aristas y se denota por k_n

Los k_n son grafos simples de n vértices, donde cada vértice es adyacente a todos los demás.

Para que un grafo completo fuera desconexo habría que eliminar todos los vértices.



$$M_a = \begin{array}{c} \begin{matrix} & A & B & C \\ A & 0 & 1 & 1 \\ B & 1 & 0 & 1 \\ C & 1 & 1 & 0 \end{matrix} \\ \hline \end{array}$$



$$M_a = \begin{array}{c} \begin{matrix} & A & B & C & D \\ A & 0 & 1 & 1 & 1 \\ B & 1 & 0 & 1 & 1 \\ C & 1 & 1 & 0 & 1 \\ D & 1 & 1 & 1 & 0 \end{matrix} \\ \hline \end{array}$$

GRAFO BIPARTIDO

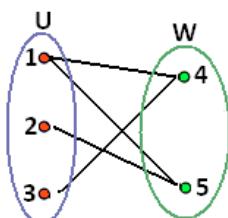
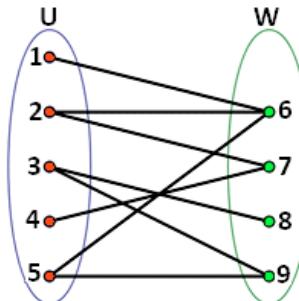
Es un grafo $G = (V, A, \varphi)$ cuyos vértices se pueden separar en dos conjuntos disjuntos U y W , es decir, tal que se verifica que $U \cup W = V$ y $U \cap W = \emptyset$. De este modo, las aristas sólo pueden conectar vértices de un conjunto con vértices del otro.

En el grafo $V = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$

$$U = \{1, 2, 3, 4, 5\} \quad W = \{6, 7, 8, 9\}$$

donde $U \cup W = V$ y $U \cap W = \emptyset$

Todas las aristas que hay, tienen un extremo en U y el otro en W



$$V = \{1, 2, 3, 4, 5\}, U = \{1, 2, 3\}, W = \{4, 5\}$$

Un Grafo Bipartido no exige que tenga que existir arista entre todo par de vértices (uno de U y el otro de W), solo exige que las aristas que existan deben de estar comprendidas entre un vértice de cada subconjunto.

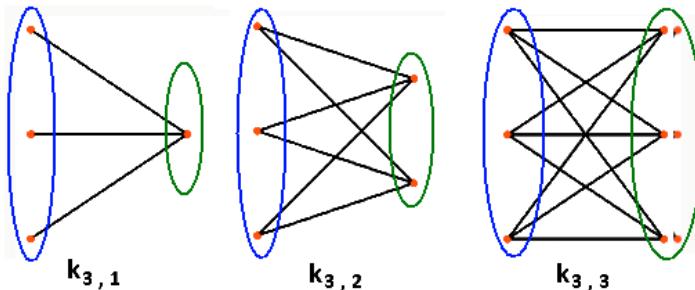
No hay una arista entre 2 y 4, cuando podía haber ocurrido.

PROPIEDADES GRAFO BIPARTIDO

- Un grafo G es bipartito si y solo si no tiene ciclos de longitud impar.
- Un grafo es bipartido si sus vértices pueden colorearse utilizando dos colores de forma que no exista ninguna arista que conecte dos vértices del mismo color.
- Cuando los dos subconjuntos U y W tienen la misma cantidad de vértices o cardinalidad, el grafo bipartido G es balanceado.
- Si todos los vértices del mismo lado de la bipartición tienen el mismo grado, G se llama grafo birregular.
- En general K_n no es bipartido, porque tres vértices están conectados entre sí y forman un ciclo de grado impar.

GRAFO BIPARTIDO COMPLETO

Un grafo bipartido $G = (V, A, \phi)$, es decir, que está formado por dos conjuntos disjuntos de vértices U y W , es *completo* cuando todas las posibles aristas unen esos vértices.



El grafo completo bipartito G con particiones de tamaño $|U| = n$ y $|W| = m$ se denota por $k_{n,m}$

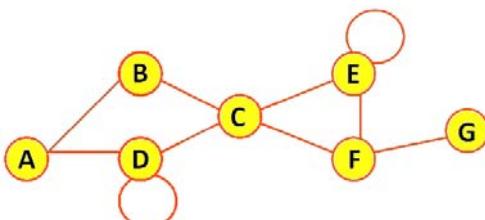
SUBGRAFO

Un subgrafo de un grafo $G = (V, A, \phi)$ es un grafo $G^* = (V^*, A^*, \phi_{A^*})$ tal que $V^* \subseteq V$, $A^* \subseteq A$, ϕ_{A^*} es la función ϕ restringida a A^* .

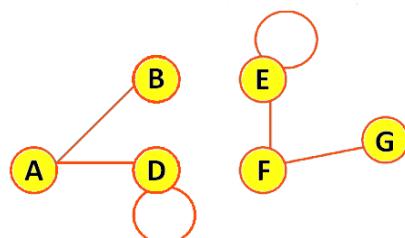
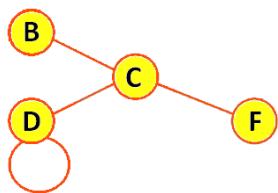
Un subgrafo se obtiene:

- Suprimiendo uno o varios vértices y las aristas incidentes en ellos. Al suprimir un vértice el subgrafo resultante es \tilde{G}_v
- Suprimiendo solamente una o varias aristas. Al suprimir una arista el subgrafo resultante es \tilde{G}_a

Sea el grafo $G = (V, A, \phi)$



Un subgrafo $G_1 = \tilde{G}_H$ se obtiene al eliminar los vértices $H = \{A, E, G\}$

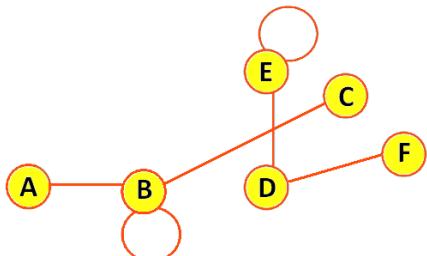
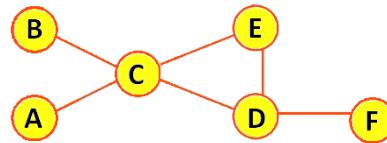


Subgrafo $G_2 = \tilde{G}_C$ obtenido al eliminar el vértice C , interrumpiendo la comunicación.

En el diseño de redes se trata de evitar puntos de corte, esto es, nodos que si tienen algún problema de funcionamiento interrumpen la comunicación entre los otros.

GRAFO CONEXO

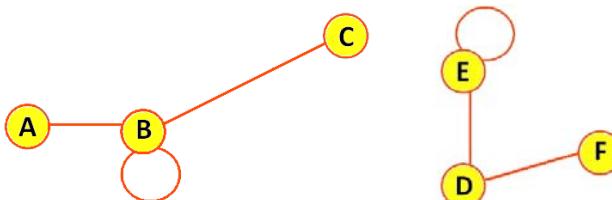
Un grafo es conexo cuando de cualquier vértice se puede llegar a cualquier otro a través de un camino.



Grafo no conexo ya que entre los vértices A y F no existe ningún camino.

No obstante, como se refleja debajo, está formado por dos subgrafos conexos, conocidos como componentes conexas.

Componentes Conexas:

**RELACIÓN DE CONEXIÓN**

Dado un grafo $G = (V, A, \varphi)$, en el conjunto de vértices $V = \{v_1, v_2, v_3, \dots, v_n\}$ se define la relación:

$$v_i R v_j \Leftrightarrow \text{Existe un camino de } v_i \text{ a } v_j \text{ o bien } v_i = v_j$$

La relación es de equivalencia, en consecuencia, pueden hallarse las clases de equivalencia, a las que se denomina componentes conexas.

ISTMO O PUNTO DE CORTE

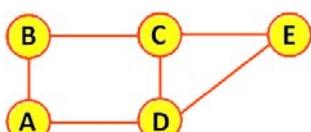
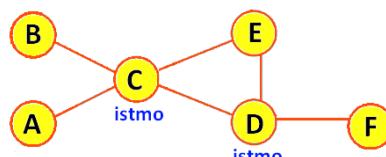
Dado un grafo conexo $G = (V, A, \varphi)$, un vértice o nodo $v \in V$ es un Istmo $\Leftrightarrow \tilde{G}_v$ es no conexo.

Es decir, un Istmo es un vértice que si se suprime desconecta el grafo.

CONECTIVIDAD: Es el menor número de vértices cuya supresión desconecta al grafo.

El grafo conexo tiene conectividad = 1 pues al suprimir el vértice C o el vértice D queda no conexo.

Los vértices C y D son istmos.



El grafo conexo tiene conectividad = 2 ya que se necesita suprimir dos vértices para que el subgrafo restante sea no conexo.

Sean los vértices C y D los vértices que se suprimen.

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

PUENTE

Dado un grafo conexo $G = (V, A, \varphi)$ con un conjunto de vértices $V = \{v_1, v_2, v_3, \dots, v_n\}$ y un conjunto de aristas $A = \{a_1, a_2, a_3, \dots, a_m\}$, se define:

$$a \in A \text{ es un Puente} \Leftrightarrow \tilde{G}_a \text{ es no conexo.}$$

Es decir, un Puente es una arista tal que su supresión desconecta al grafo.

Una arista es un Puente \Leftrightarrow No está contenida en ningún ciclo.

Un grafo sin puentes equivale a un grafo conexo con conectividad 2

GRAFO EULERIANO

Un camino de Euler es una trayectoria que contiene todas las aristas de G y recorre cada arista una sola vez.

Si un grafo tiene un camino euleriano, se dice que el grafo es euleriano

Condición necesaria y suficiente para que en un grafo exista un camino euleriano:

- El grafo debe ser conexo.
- Todos los vértices deben tener grado par, o a lo sumo dos vértices con grado impar.

En general k_n para n par y $n > 2$ no es euleriano porque todos sus vértices tienen grado $(n - 1)$ impar.

CICLO EULERIANO

Cuando el camino euleriano comienza y termina en el mismo vértice se denomina ciclo euleriano (circuito ó camino cerrado).

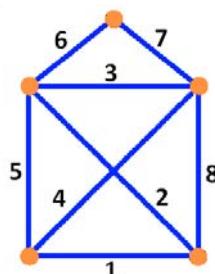
Es decir, un ciclo euleriano es un ciclo que pasa por todas las aristas una sola vez.

Condición necesaria y suficiente para que en un grafo exista un ciclo euleriano:

- El grafo debe ser conexo.
- Todos los vértices deben tener grado par.

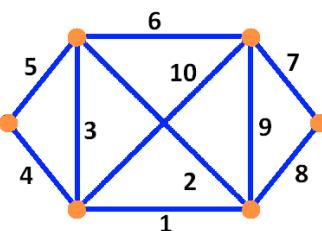
El grafo es un camino euleriano.

No es ciclo euleriano porque hay dos vértices de grado 3

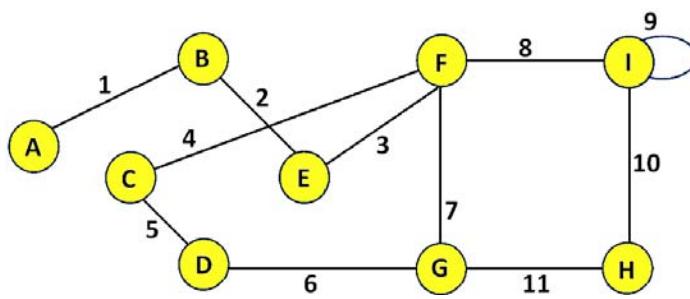
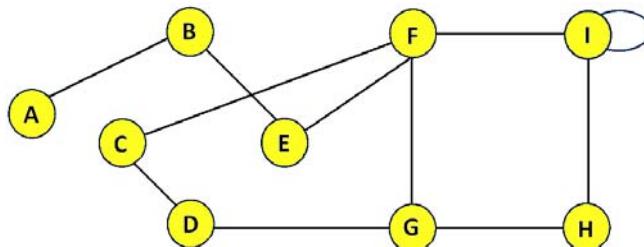


Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

El grafo tiene un ciclo euleriano porque todos sus vértices tienen grado par.



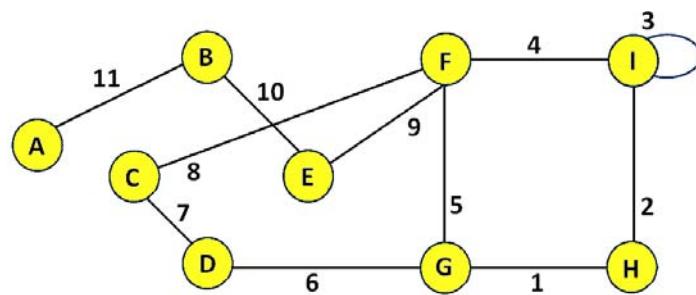
Encontrar un camino euleriano en el grafo:



El camino euleriano no es único.

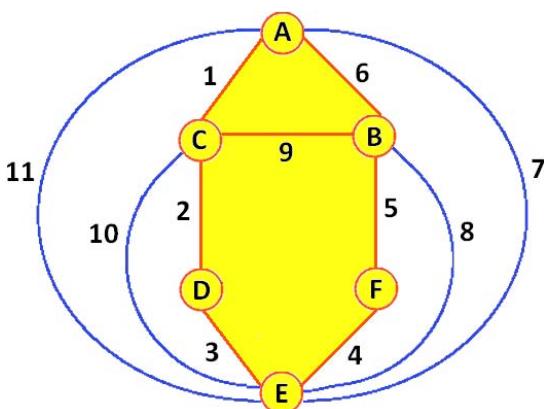
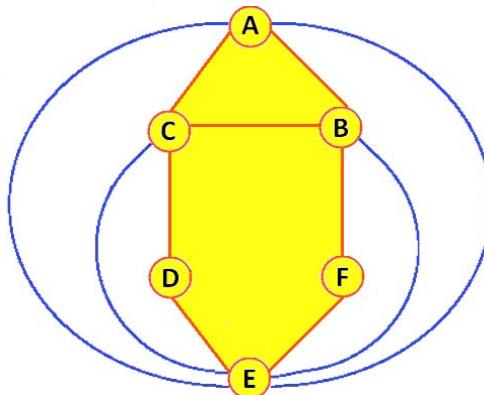
Camino Euleriano 1

Camino Euleriano 2



Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

- Encontrar un ciclo euleriano en el grafo:

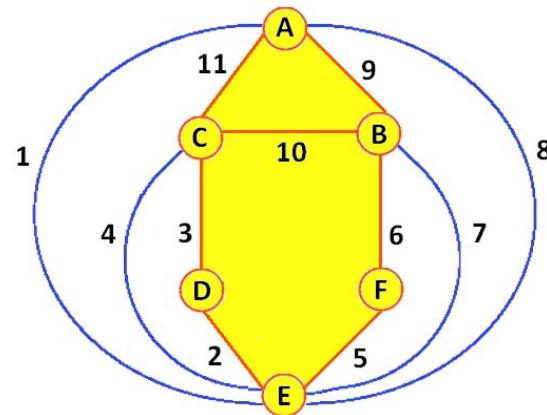


Un ciclo de Euler C_1

$$C_1 = \{A, 1; C, 2; D, 3; E, 4; F, 5; B, 6; A, 7; E, 8; B, 9; C, 10; E, 11; A\}$$

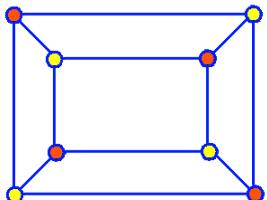
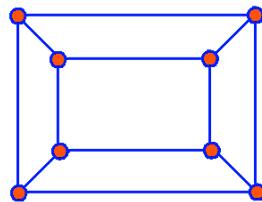
Otro ciclo de Euler C_2

$$C_2 = \{A, 1; E, 2; D, 3; C, 4; E, 5; F, 6; B, 7; E, 8; A, 9; B, 10; C, 11; A\}$$



Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

- Analizar si el grafo adjunto es euleriano y bipartido.



El grafo no es euleriano porque los grados de los vértices no son todos pares.

Un grafo es bipartido si sus vértices pueden colorearse utilizando dos colores de forma que no exista ninguna arista que conecte dos vértices del mismo color.

Es bipartido porque se puede colorear con dos colores.

CAMINO HAMILTONIANO

Sea un grafo con $|V| \geq 3$ sin vértices aislados (con grado cero; es decir, un vértice que no es punto final de ninguna arista).

Un camino hamiltoniano pasa una sola vez por cada vértice.

No es necesario que pase por todas las aristas, pues en muchos casos repetiría vértices y no sería camino hamiltoniano.

Si ese camino es cerrado se denomina Ciclo Hamiltoniano.

Si un grafo contiene un Ciclo Hamiltoniano se conoce como Grafo Hamiltoniano.

Condiciones suficientes para caracterizar Grafos Hamiltonianos:

Teorema de Dirac: Si para todo vértice v_i de un grafo simple $G = (V, A, \varphi)$, con $n \geq 3$ vértices se verifica que $g(v_i) \geq n/2 \Rightarrow G$ es un grafo hamiltoniano.

Teorema de Ore: Sea un grafo simple $G = (V, A, \varphi)$, con $n \geq 3$ vértices. Si se verifica que $g(v) + g(w) \geq n \quad \forall v, w \in V$ con $v \neq w$ y no adyacentes, entonces el grafo G posee un ciclo hamiltoniano.

Corolario 1: Un grafo simple $G = (V, A, \varphi)$, con $n \geq 2$ vértices, si verifica: $g(v_i) \geq \frac{n-1}{2} \rightarrow G$ tiene un camino hamiltoniano.

Corolario 2: Un grafo simple $G = (V, A, \varphi)$, con $n \geq 3$ vértices tiene un ciclo hamiltoniano si

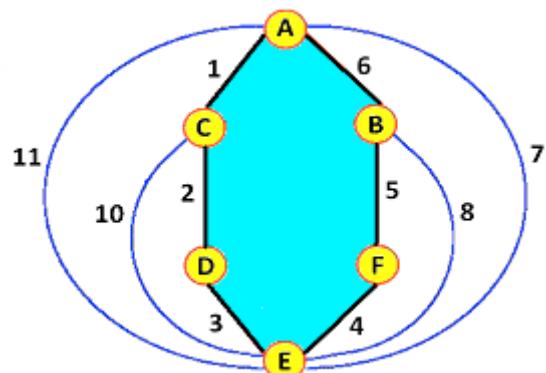
$$|A| \geq \binom{n-1}{2} + 2$$

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

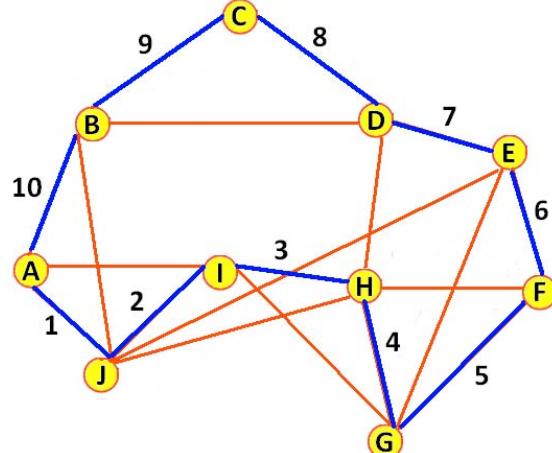
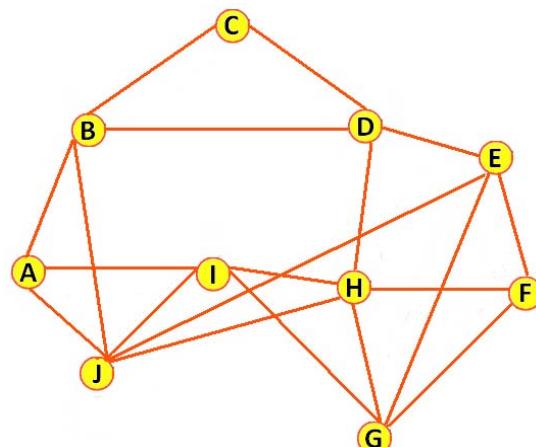
En el grafo anterior, un posible ciclo hamiltoniano sería:

$$C = \{A, 1; C, 2; D, 3; E, 4; F, 5; B, 6; A\}$$

Un camino hamiltoniano pasa una sola vez por cada vértice.



Encontrar un camino hamiltoniano en el grafo:



Un posible ciclo hamiltoniano es:

$$C_1 = \{A, 1; J, 2; I, 3; H, 4; G, 5; F, 6; E, 7; D, 8; C, 9; B, 10; A\}$$

Un camino hamiltoniano pasa una sola vez por cada vértice.

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

- Dado el grafo G con matriz de adyacencia

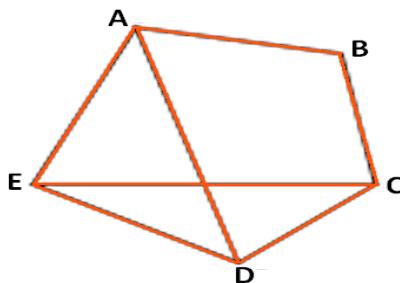
Analizar si

<input checked="" type="checkbox"/> Es Conexo <input checked="" type="checkbox"/> Es Euleriano <input checked="" type="checkbox"/> Es un Multigrafo

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

Considerando el conjunto de vértices $\{A, B, C, D, E\}$, el grado de cada vértice se obtiene sumando los elementos de cada fila.

$$M = \begin{matrix} & \textbf{A} & \textbf{B} & \textbf{C} & \textbf{D} & \textbf{E} \\ \textbf{A} & \left(\begin{matrix} 0 & 1 & 0 & 1 & 1 \end{matrix} \right) & \rightarrow & gr(A) = 3 \\ \textbf{B} & \left(\begin{matrix} 1 & 0 & 1 & 0 & 0 \end{matrix} \right) & \rightarrow & gr(B) = 2 \\ \textbf{C} & \left(\begin{matrix} 0 & 1 & 0 & 1 & 1 \end{matrix} \right) & \rightarrow & gr(C) = 3 \\ \textbf{D} & \left(\begin{matrix} 1 & 0 & 1 & 0 & 1 \end{matrix} \right) & \rightarrow & gr(D) = 3 \\ \textbf{E} & \left(\begin{matrix} 1 & 0 & 1 & 1 & 0 \end{matrix} \right) & \rightarrow & gr(E) = 3 \end{matrix}$$



- Representando el Grafo asociado a la matriz de adyacencia se observa que es Conexo, es decir, todos los vértices están conectados a través de un camino. No hay ningún vértice aislado de los demás.
- No es un Grafo Euleriano porque tiene 4 vértices con grado impar, para que sea un Grafo Euleriano tienen que ser todos de grado par (a lo sumo dos vértices con grado impar).
- Un Multigrafo o Pseudografo es un grafo que está facultado para tener aristas múltiples; es decir, aristas que relacionan los mismos nodos.

Para que sea un Multigrafo debe haber en la matriz de adyacencia más de un número superior a 1, indicando con estos números que dos vértices se conectan con más de una arista.

En consecuencia, no es Multigrafo.

- Dada la matriz de adyacencia del grafo G, analizar si es un pseudografo.

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

$$M = \begin{pmatrix} \overbrace{m_{11}=1}^{\text{Lazo}} & 1 & 1 & 0 \\ 1 & \overbrace{m_{22}=1}^{\text{Lazo}} & 0 & 1 \\ 1 & 0 & \overbrace{m_{33}=1}^{\text{Lazo}} & 1 \\ 0 & 1 & 1 & \overbrace{m_{44}=1}^{\text{Lazo}} \end{pmatrix}$$

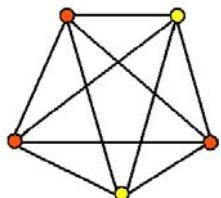
En los Pseudografos o Multigrafos se permiten Lazos o Buclees, es decir, aristas cuyos extremos coinciden.

Se trata de un Pseudografo

Analizar si el grafo adjunto es euleriano, bipartido y pseudografo.



- El grafo es conexo (desde cualquier vértice se puede llegar a cualquier otro vértice a través de un camino) y todos los vértices tienen grado par, luego es un grafo euleriano.
- Un grafo bipartido no tiene ciclos de longitud impar. El grafo tiene ciclos de longitud impar (3 y 5), luego no es bipartido.



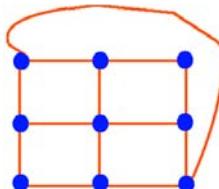
De otra parte, un grafo es bipartido si sus vértices pueden colorearse utilizando dos colores de forma que no exista ninguna arista que conecte dos vértices del mismo color.

Se observa que no es bipartido.

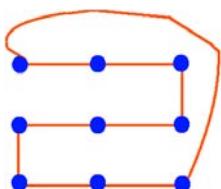
- Un multigrafo o pseudografo es un grafo que contiene aristas paralelas, aristas con los mismos nodos o vértices iniciales y finales. En esta línea, no es un pseudografo porque no tiene vértices con aristas origen y extremo final.

Analizar si el grafo es:

Euleriano
 Hamiltoniano
 Bipartido



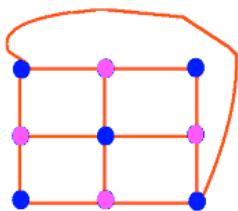
No es un grafo Euleriano porque hay seis vértices que tienen grado 3, cuando en un grafo Euleriano se permite a lo sumo dos vértices de grado impar.



Un grafo que contiene un ciclo Hamiltoniano se denomina grafo Hamiltoniano.

El subgrafo de la figura contiene un ciclo Hamiltoniano (cerrado y pasa una vez por cada vértice)

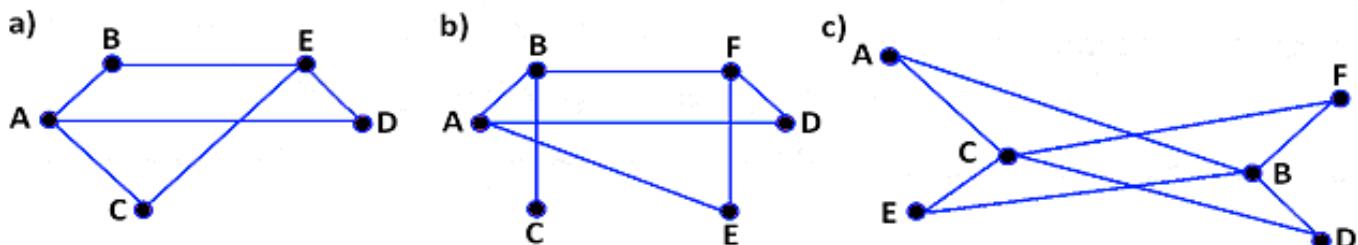
Un grafo es bipartito (o bipartido) no tiene ciclos de longitud impar. En consecuencia, no es bipartido porque el ciclo asociado a la figura es de longitud impar.



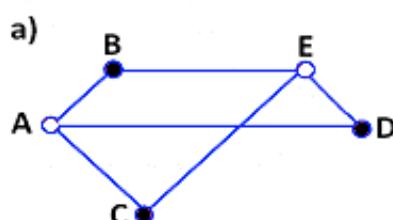
Por otro lado, un grafo es bipartido si sus vértices pueden colorearse utilizando dos colores de forma que no exista ninguna arista que conecte dos vértices del mismo color.

Por tanto, no es bipartido.

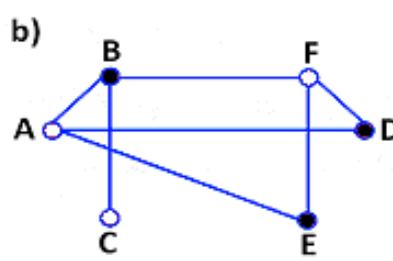
Determinar qué grafos son bipartidos:



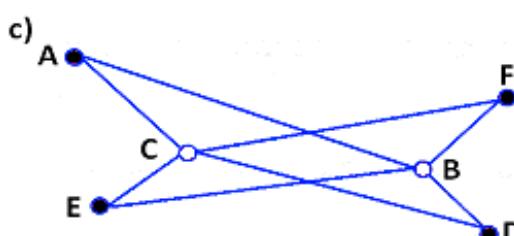
Un grafo es bipartido si sus vértices pueden colorearse utilizando dos colores de forma que no exista ninguna arista que conecte dos vértices del mismo color.



Es bipartido, la partición de $V = \{A, B, C, D, E\}$ tal que $V_1 = \{B, D, C\}$ y $V_2 = \{A, E\}$, siendo $V = V_1 \cup V_2$ y $V_1 \cap V_2 = \emptyset$



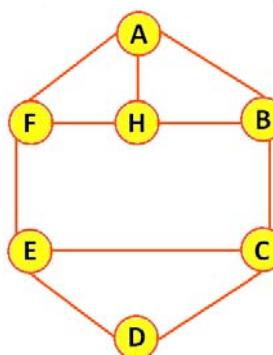
Es bipartido, la partición de $V = \{A, B, C, D, E, F\}$ es $V_1 = \{B, D, E\}$ y $V_2 = \{A, C, F\}$, siendo $V = V_1 \cup V_2$ y $V_1 \cap V_2 = \emptyset$



Es bipartido, la partición de $V = \{A, B, C, D, E, F\}$ es $V_1 = \{A, D, E, F\}$ y $V_2 = \{C, B\}$, siendo $V = V_1 \cup V_2$ y $V_1 \cap V_2 = \emptyset$

Analizar si el grafo es:

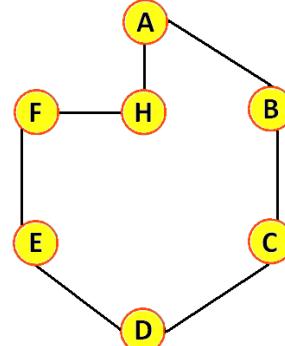
Euleriano
 3 regular
 Hamiltoniano



- a) No puede ser un grafo Euleriano porque todos los vértices son de grado 3 excepto el vértice D que es de grado 2.

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

- b) No puede ser un grafo 3-regular porque todos los vértices tendrían que tener grado 3, siendo el vértice D de grado 2.

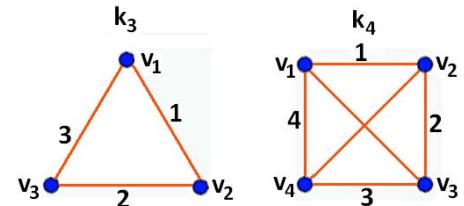


- c) Es un grafo hamiltoniano porque el camino del subgrafo es un ciclo hamiltoniano al ser un camino cerrado, que pasa por todos los vértices una sola vez.

Sea el grafo completo k_n (n par, $n \geq 3$) es:
 Euleriano
 Hamiltoniano
 Bipartido

Se analizan dos grafos completos para decidir por las opciones.

Los grafos completos se indican por k_n , grafos simples de n vértices, en donde cada vértice es adyacente a todos los demás vértices.



- a) Para que un grafo sea euleriano debe ser conexo y todos los vértices deben de tener grado par o la suma de los vértices con grado impar.

k_3 es euleriano y k_4 no es euleriano

En general k_n para n par y $n > 2$ no es euleriano porque todos sus vértices tienen grado $(n - 1)$ impar.

- b) Los dos grafos completos son hamiltonianos (pasa una sola vez por cada vértice), basta escoger los ciclos:

$$k_3 : C = \{v_1, 1; v_2, 2; v_3, 3; v_1\}$$

$$k_4 : C = \{v_1, 1; v_2, 2; v_3, 3; v_4, 4; v_1\}$$

- c) En general k_n no es bipartido, porque tres vértices están conectados entre sí y forman un ciclo de grado impar.

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

ISOMORFISMOS DE GRAFOS: Dados dos grafos $G_1 = (V_1, A_1, \varphi_1)$ y $G_2 = (V_2, A_2, \varphi_2)$ se dice que son isomorfos si y solo si existen dos funciones biyectivas:

$$f: V_1 \longrightarrow V_2 \quad y \quad g: A_1 \longrightarrow A_2 \text{ tales que } \varphi_2[g(a)] = f[\varphi_1(a)] \quad \forall a \in A_1$$

Cuando no hay aristas paralelas es suficiente que $\forall u, v \in V_1: \{u, v\} \in A_1 \Rightarrow \{f(u), f(v)\} \in A_2$

Es decir, si en el primer grafo hay una arista entre dos vértices, los vértices correspondientes en el segundo grafo también deben de estar unidos por una arista.

En otras palabras, dos grafos son isomorfos cuando tienen la misma estructura, esto es, sus vértices están relacionados de igual forma aunque estén dibujados de forma distinta.

◆ Condiciones necesarias de Isomorfismo entre dos grafos:

La misma cantidad de vértices

La misma cantidad de aristas

Los mismos grados de los vértices

Cadenas de las mismas longitudes

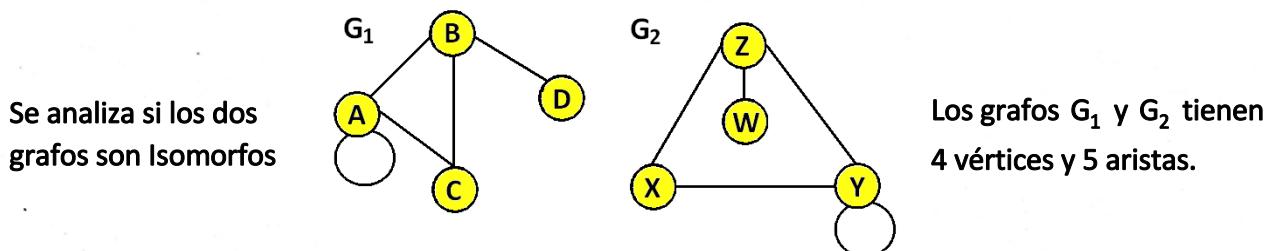
Si un grafo tiene ciclos, el otro grafo también debe tenerlos

◆ Condición suficiente de Isomorfismo entre dos grafos:

Los dos grafos tienen la misma matriz de adyacencia

Si las matrices de adyacencia correspondientes a dos grafos no son iguales no se puede concluir que los dos grafos no sean isomorfos, cabe la posibilidad de que reordenando una de las matrices de adyacencia se pueda conseguir que sean iguales.

En la práctica, para afirmar que dos grafos no son isomorfos hay que mostrar alguna propiedad estructural no compartida.



Se define la función biyectiva haciendo corresponder los vértices con iguales grados:

$$f(A) = Y \quad f(B) = Z \quad f(C) = X \quad f(D) = W$$

Si hay una arista entre dos vértices del primer grafo G_1 también debe haber una arista entre los vértices correspondientes del grafo G_2 .

Se observa que entre A y B hay una arista en G_1 y también hay una arista entre $Y = f(A)$ y $Z = f(B)$ en G_2 . Operación que habría que realizar para cada arista.

Se puede comprobar para todas las aristas juntas con la matriz de adyacencia ordenando convenientemente los vértices, de acuerdo con la función biyectiva definida entre los vértices:

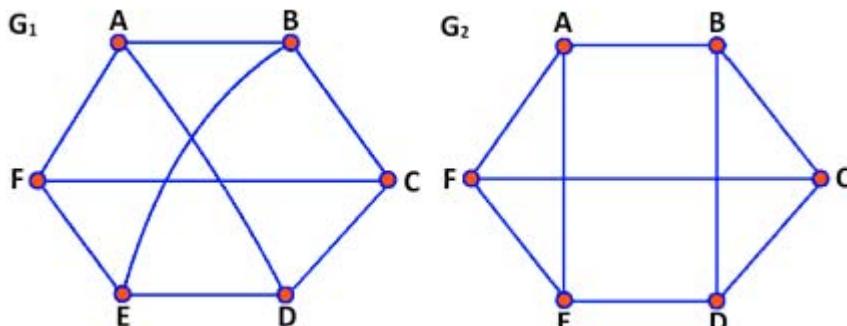
$$G_2 = \begin{array}{c|cccc} & Y & Z & X & W \\ \hline Y & 1 & 1 & 1 & 0 \\ Z & 1 & 0 & 1 & 1 \\ X & 1 & 1 & 0 & 0 \\ W & 0 & 1 & 0 & 0 \end{array} \quad G_1 \equiv \begin{array}{c|cccc} & A & B & C & D \\ \hline A & 1 & 1 & 1 & 0 \\ B & 1 & 0 & 1 & 1 \\ C & 1 & 1 & 0 & 0 \\ D & 0 & 1 & 0 & 0 \end{array}$$

Como las matrices de adyacencia de G_1 y G_2 son iguales se puede afirmar que G_1 es isomorfo a G_2

- Se dispone de 6 ordenadores y 9 cables de conexión. Se pretende que cada ordenador se conecte con otros tres. ¿Existe alguna forma de conectarlos?. ¿Hay más de una forma?.

Para representar el ejercicio se utiliza un grafo de 6 vértices (ordenadores) y 9 aristas (cables de conexión). Se pregunta: ¿Existe un grafo con 6 vértices, 9 aristas y que sea regular de grado 3 que verifique las condiciones?.

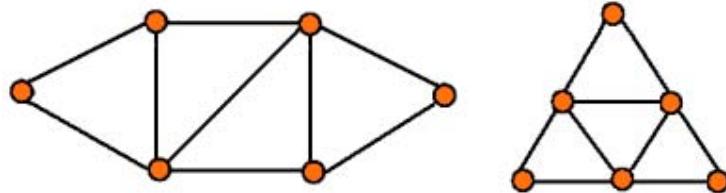
Los grafos G_1 y G_2 cumplen las condiciones. En consecuencia, existen al menos de conectar 6 ordenadores.



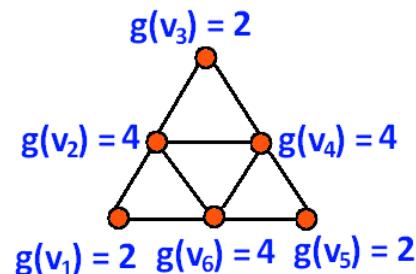
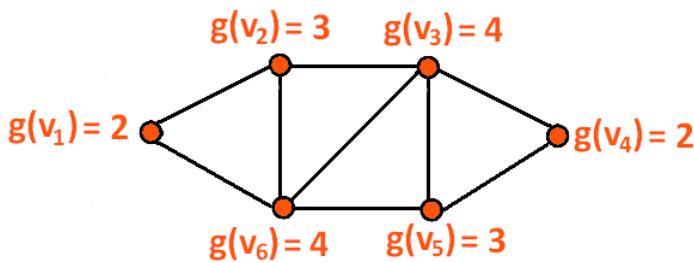
G_1 y G_2 no son isomorfos, dado que en G_2 hay 3 ciclos y no existen en G_1 (tienen que tener la misma estructura).

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

¿Los grafos son isomorfos?



Para que dos grafos sean isomorfos tienen que conservar la misma estructura.



No son isomorfos pues en el primer grafo hay dos vértices de grado 2 mientras que en el segundo grafo hay tres vértices de grado 2.

Si hubiera un Isomorfismo se tendría que conservar el grado de los vértices pues no se pueden hacer corresponder biunívocamente los vértices de grado 2.

Dadas las matrices de adyacencia de tres grafos, establecer el isomorfismo de los grafos.

$$M_a(G_A) = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix} \quad M_a(G_B) = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \quad M_a(G_C) = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$$

Para que dos grafos sean isomorfos tienen que conservar la misma estructura, analizando los grados de los vértices:

$$M_a(G_A) = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix} \rightarrow g(v_1)=1 \quad M_a(G_B) = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \rightarrow g(w_1)=3$$

$$\rightarrow g(v_2)=3 \quad \rightarrow g(w_2)=1$$

$$\rightarrow g(v_3)=2 \quad \rightarrow g(w_3)=2$$

$$\rightarrow g(v_4)=2 \quad \rightarrow g(w_4)=2$$

$$M_a(G_C) = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix} \rightarrow g(z_1)=3$$

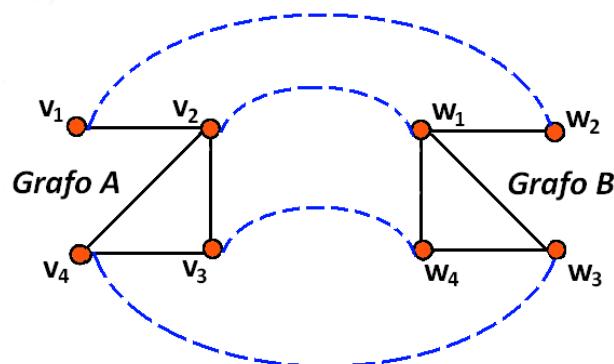
$$\rightarrow g(z_2)=3 \quad \rightarrow g(w_1)=3$$

$$\rightarrow g(z_3)=3 \quad \rightarrow g(w_2)=1$$

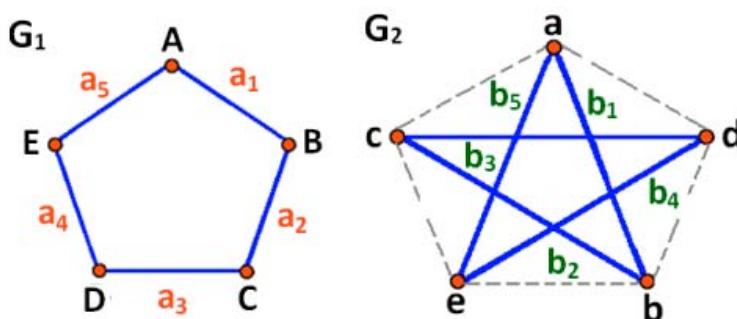
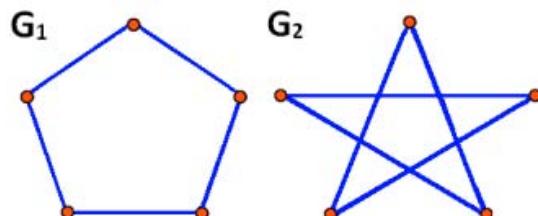
$$\rightarrow g(z_4)=3 \quad \rightarrow g(w_3)=2$$

No se puede establecer un isomorfismo entre G_A y el grafo G_C , ni entre el grafo G_B y el grafo G_C , dado que no se pueden hacer corresponder el grado de los vértices.

Entre el grafo G_A y G_B se pueden hacer corresponder los vértices:



¿Hay un isomorfismo entre los grafos?



El isomorfismo entre los grafos G_1 y G_2 está definido por

$$\begin{cases} f(A) = a \\ f(B) = b \\ f(C) = c \\ f(D) = d \\ f(E) = e \end{cases} \quad y \quad g(a_i) = b_i \quad i = 1, 2, \dots, 5$$

Los grafos G_1 y G_2 son isomorfos solo y solo si para alguna ordenación de vértices y lados sus matrices de incidencia son iguales (conservan la misma estructura).

$$M_i(G_1) = \begin{matrix} & a_1 & a_2 & a_3 & a_4 & a_5 \\ A & (1 & 0 & 0 & 0 & 1) \\ B & (1 & 1 & 0 & 0 & 0) \\ C & (0 & 1 & 1 & 0 & 0) \\ D & (0 & 0 & 1 & 1 & 0) \\ E & (0 & 0 & 0 & 1 & 1) \end{matrix} \quad M_i(G_2) = \begin{matrix} & b_1 & b_2 & b_3 & b_4 & b_5 \\ a & (1 & 0 & 0 & 0 & 1) \\ b & (1 & 1 & 0 & 0 & 0) \\ c & (0 & 1 & 1 & 0 & 0) \\ d & (0 & 0 & 1 & 1 & 0) \\ e & (0 & 0 & 0 & 1 & 1) \end{matrix}$$

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

ELEMENTOS DE UN GRAFO: Sea el grafo $G = (V, A, \varphi)$,

DISTANCIA ENTRE VÉRTICES: Mínima longitud de los caminos que unen dos vértices:

$$d(v_i, v_j) = \min \{ \text{long}(v_i, v_j) \}$$

Cuando no existe camino entre los vértices $d(v_1, v_2) = \infty$

EXCENTRICIDAD DE UN VÉRTICE: Máxima distancia entre un vértice v_i con cualquier otro vértice:

$$\text{ex}(v_i) = \max \{ d(v_i, v_j) \} \quad \forall v_j \in V$$

RADIO DEL GRAFO: Mínima excentricidad de los vértices: $\text{rad}(G) = \min \{ \text{e}(v_i) \} \quad \forall v_i \in V$

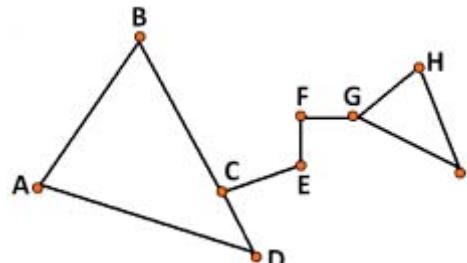
DIÁMETRO DEL GRAFO: Máxima excentricidad de los vértices:

$$\text{diam}(G) = \max \{ \text{e}(v_i) \} \quad \forall v_i \in V$$

VÉRTICES CENTRALES DEL GRAFO: Vértices con igual excentricidad que el radio.

CENTRO DE UN GRAFO: Subgrafo inducido por el conjunto de vértices de mínima excentricidad.

Sea el grafo $G = (A, B, C, D, E, F, G, H, I)$



La excentricidad del vértice A al vértice I: $e(A) = 6$

$$\text{ex}(A) = \text{ex}(H) = \text{ex}(I) = 6$$

Haciendo lo mismo con los demás vértices del grafo:

$$\begin{aligned} \text{ex}(B) &= \text{ex}(D) = \text{ex}(G) = 5 \\ \text{ex}(C) &= \text{ex}(F) = 4 \\ \text{ex}(E) &= 3 \end{aligned} \rightarrow \begin{aligned} \text{rad}(G) &= 3 \\ \text{diam}(G) &= 6 \end{aligned}$$

Vértice central: $\text{rad}(G) = 3 = \text{ex}(E) \rightarrow v_c = E$

Centro del grafo: $C(G) = \{E\}$

- Calcular la excentricidad de los vértices, radio, diámetro del grafo, vértices centrales y centro.



$$\begin{aligned} \text{ex}(A) = \text{ex}(E) &= 4 \\ \text{ex}(B) = \text{ex}(D) &= 3 \rightarrow \text{rad}(G) = 2 \\ \text{ex}(C) &= 2 \quad \text{diam}(G) = 4 \end{aligned}$$

Vértice central: $\text{rad}(G) = 2 = \text{ex}(C) \rightarrow v_c = C$

Centro del grafo: $C(G) = \{C\}$

GRAFO ORIENTABLE: Cuando en un grafo se asigna un sentido a cada una de las aristas se obtiene una orientación del grafo.

Un ejemplo clásico ocurre con el callejero de una localidad, donde es necesario poder ir desde un punto origen a cualquier punto destino.

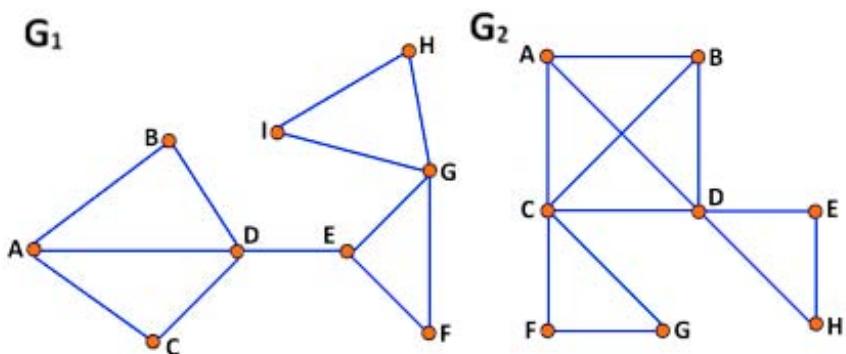
El problema que se plantea, partiendo del grafo, equivale a la obtención de un dígrafo fuertemente conexo. Cuando la construcción del dígrafo es posible se dice que el grafo es orientable.

Un grafo G es orientable si y sólo si es conexo y no posee puentes (arista que su supresión desconecta el grafo).

El algoritmo para orientar un grafo de Hopcroft y Tarjan resuelve la cuestión:

- ◆ Se comienza con un vértice al que se le asigna la etiqueta 1
- ◆ Se considera el vértice v_i con la mayor etiqueta "t" y tal que tenga un vértice adyacente v_j aún sin etiquetar. Se orienta la arista a_{ij} del vértice v_i a v_j , asignando a v_j la etiqueta "t+1"
- ◆ Cuando todos los vértices sean etiquetados, se orientan las aristas restantes siempre desde el vértice con etiqueta superior al vértice con etiqueta inferior.

- Determinar la orientación de los grafos:



Orientable significa que se puede dar sentido a las aristas, resultando un dígrafo donde existe un camino dirigido entre cada par de vértices.

El grafo G_1 no es orientable porque tiene un Puente o Istmo, una arista que al ser eliminada desconecta el grafo.

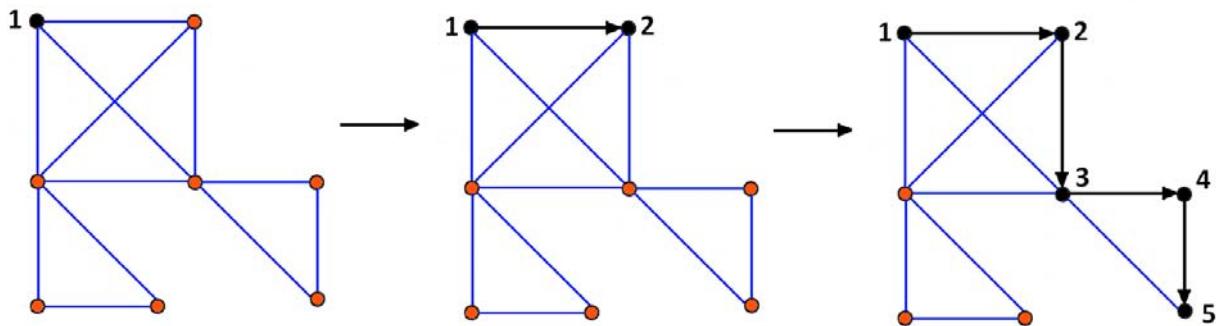
Una arista es un Puente \Leftrightarrow No está contenida en ningún ciclo.

Un grafo sin puentes equivale a un grafo conexo con conectividad 2.

El grafo G_2 es un grafo conexo y sin puentes, en consecuencia es un grafo orientable.

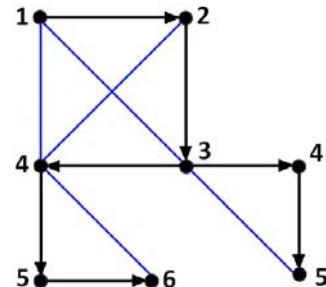
Utilizando el algoritmo de Hopcroft y Tarjan se parte de un vértice cualquiera que se etiqueta como "1", se toma un vértice adyacente etiquetado como "2" y se orienta la arista de 1 a 2.

Se sigue con el proceso, etiquetando vértices y orientando aristas hasta que todos los vértices adyacentes se encuentren etiquetados, hasta llegar al último vértice etiquetado "5".

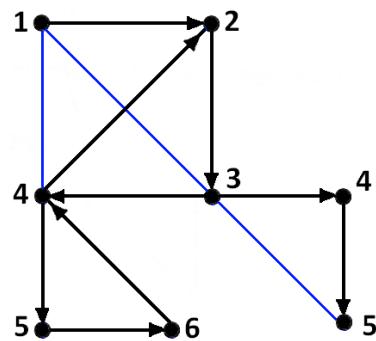


Se retrocede por el camino seguido hasta encontrar un vértice con algún vértice adyacente sin etiquetar.

De este modo, se encuentra el vértice con etiqueta "3" etiquetando el vértice adyacente con 4 y orientando la arista.

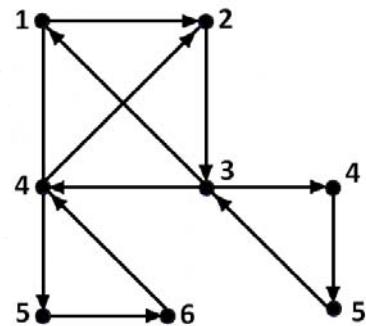


El proceso continúa hasta etiquetar todos los vértices, la etiqueta que recibe cada vértice es una unidad superior a la del vértice adyacente del que proviene.



Las aristas que no han sido orientadas en el proceso de etiquetado se orientan del vértice de mayor etiqueta al vértice de menor etiqueta.

Finalmente, se obtiene la orientación del grafo inicial, un dígrafo fuertemente conexo



COMPONENTE FUERTEMENTE CONEXO: Un grafo dirigido se dice que es fuertemente conexo si para cada par de vértices (u_i, u_j) existe un camino de u_i hacia v_j y un camino de v_j hacia u_i .

Los componentes fuertemente conexos (CFC) de un grafo dirigido son sus subgrafos maximales fuertemente conexos. Estos subgrafos forman una partición del grafo.

GRAFOS, DÍGRAFOS Y ÁRBOLES: En muchas aplicaciones es necesario indicar el camino de las aristas, surgen los dígrafos o grafos dirigidos.

DÍGRAFO: Es una terna $G = (V, A, \delta)$, donde V es un conjunto no vacío de vértices, A es un conjunto de

aristas o arcos y δ es la función de incidencia: $\delta : A \longrightarrow V \times V$

La función de incidencia se dice dirigida.

La función de incidencia δ hace corresponder a cada arista un "par ordenado" de vértices, al primero se le conoce como "extremo Inicial" y al segundo como "vértice final".

Los caminos y ciclos se definen de la misma forma que con los grafos no dirigidos, respetando el sentido de las aristas.

Cuando todos los vértices son distintos se trata de un Camino Simple.

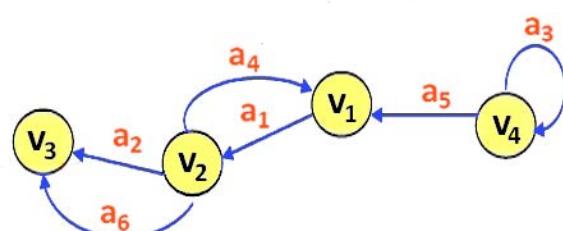
Si todas las aristas son distintas, se trata de un Camino Elemental.

Sea un grafo G con $V = \{v_1, v_3, v_2, v_4\}$ y $A = \{a_1, a_2, a_3, a_4, a_5, a_6\}$ tal que la función de incidencia dirigida viene dada por:

$$\delta(a_1) = (v_1; v_2) \quad \delta(a_2) = (v_2; v_3) \quad \delta(a_3) = (v_4; v_4)$$

$$\delta(a_4) = (v_2; v_1) \quad \delta(a_5) = (v_4; v_1) \quad \delta(a_6) = (v_2; v_3)$$

Hacer un diagrama del grafo indicando el extremo inicial y final de la arista a_5 , aristas paralelas y aristas antiparalelas, un camino y un ciclo.



Extremo Inicial de a_5 : v_4 Extremo Final de a_5 : v_1

Aristas paralelas: a_2 y a_6

Aristas antiparalelas: a_1 y a_4

Camino: $C = \{v_4, a_5 ; v_1, a_1 ; v_2, a_2 ; v_3\}$

Ciclo: $C = \{v_1, a_1 ; v_2, a_4 ; v_1\}$

FUNCIÓN GRADO EN UN DÍGRAFO: Algunas definiciones son:

GRADO POSITIVO: Cantidad de aristas que inciden positivamente en el vértice (aristas que "entran" al vértice), se denota por $g^-(v)$, siendo $\sum g^-(v_i) = |A|$

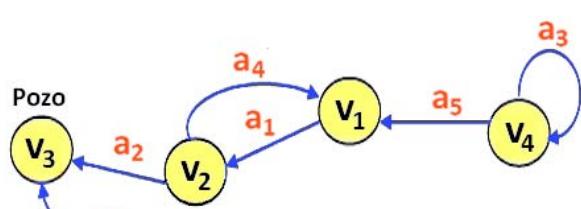
GRADO NEGATIVO: Cantidad de aristas que inciden negativamente en el vértice (aristas que "salen" del vértice), se denota por $g^+(v)$, donde $\sum g^+(v_i) = |A|$

GRADO NETO: La diferencia entre el número de aristas que entran y el número de aristas que salen, se denota por $g_N(v)$, donde $\sum g_N(v_i) = 0$

GRADO TOTAL: La suma del número de aristas que entran y el número de aristas que salen, se denota por $g(v)$, donde $\sum g(v_i) = 2|A|$

POZO: Es un vértice del que "no sale" ninguna arista: v_i es un Pozo $\Rightarrow g^+(v_i) = 0$, es decir, v_i no es extremo inicial de ninguna arista.

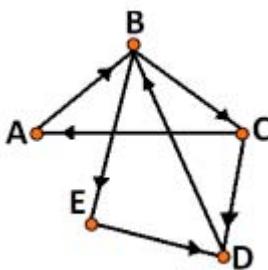
FUENTE: Es un vértice al que "no llega" ninguna arista: v_i es una Fuente $\Rightarrow g^-(v_i) = 0$, es decir, v_i no es extremo final de ninguna arista.



$$\begin{array}{llll}
 g^-(v_1) = 2 & g^+(v_1) = 1 & g_N(v_1) = 1 & g(v_1) = 3 \\
 g^-(v_2) = 1 & g^+(v_2) = 3 & g_N(v_2) = -2 & g(v_2) = 4 \\
 g^-(v_3) = 2 & g^+(v_3) = 0 & g_N(v_3) = 2 & g(v_3) = 2 \\
 g^-(v_4) = 1 & g^+(v_4) = 2 & g_N(v_4) = -1 & g(v_4) = 3
 \end{array}$$

$$\sum_{i=1}^4 v_i = 12 = 2|A| \quad A = 6 \text{ aristas}$$

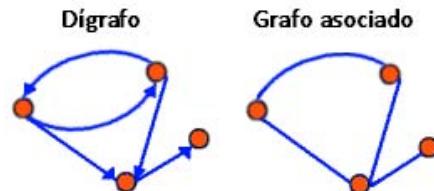
- Ejercicio:** Se considera el dígrafo $G = (V, A)$, donde $V = \{A, B, C, D, E\}$ y $A = \{AB, BC, BE, CA, CD, DB, ED\}$. Hacer una representación gráfica y calcular los grados de cada vértice.



$$\begin{array}{llll}
 g^-(A) = 1 & g^+(A) = 1 & g_N(A) = 0 & g(A) = 2 \\
 g^-(B) = 2 & g^+(B) = 2 & g_N(B) = 0 & g(B) = 4 \\
 g^-(C) = 1 & g^+(C) = 2 & g_N(C) = -1 & g(C) = 3 \\
 g^-(D) = 2 & g^+(D) = 1 & g_N(D) = 1 & g(D) = 3 \\
 g^-(E) = 1 & g^+(E) = 1 & g_N(E) = 0 & g(E) = 2
 \end{array}$$

GRAFO ASOCIADO A UN DÍGRAFO

El grado asociado a un dígrafo se obtiene al cambiar las aristas dirigidas del dígrafo por aristas no dirigidas.



CONEXIDAD EN DÍGRAFOS

Un dígrafo es conexo cuando su grafo asociado es conexo.

Un dígrafo es Fuertemente Conexo cuando existe algún camino entre todo par de vértices.



El dígrafo es conexo porque su grafo lo es.

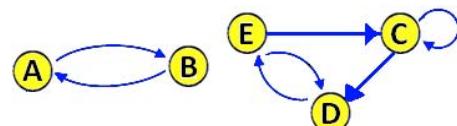
Además es fuertemente conexo.



El dígrafo es conexo porque su grafo asociado lo es.

El dígrafo no es fuertemente conexo ya que no existe camino alguno que salga del vértice E y llegue al vértice B.

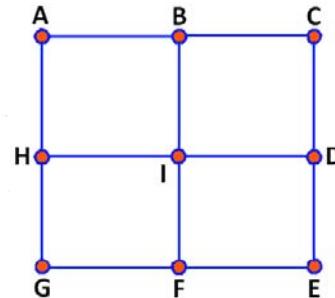
Hay componentes fuertemente conexas (CFC), es decir, para cada par de vértices existe un camino de un vértice v_i a otro v_j y un camino del vértice v_j a otro v_i



Las componentes fuertemente conexas (CFC) de un dígrafo son subgrafos maximales fuertemente conexos. Estos subgrafos forman una partición del dígrafo.

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

- Dado el grafo el grafo G , determinar la excentricidad de sus vértices, su radio y diámetro, sus vértices centrales y su centro.



$$\begin{aligned} \text{ex}(A) = \text{ex}(C) = \text{ex}(E) = \text{ex}(G) &= 4 \\ \text{ex}(B) = \text{ex}(D) = \text{ex}(F) = \text{ex}(H) &= 3 \quad \rightarrow \quad \text{rad}(G) = 2 \\ \text{ex}(I) &= 2 \end{aligned}$$

Vértice central: $\text{rad}(G) = 2 = \text{ex}(I) \rightarrow v_c = I$

Centro del grafo: $C(G) = \{I\}$

CAMINO EULERIANO - HAMILTONIANO: En la red de comunicaciones de un aeropuerto se han detectado averías, por lo que se hace necesario analizar todos los nodos para verificar las conexiones. La etiqueta de cada tramo representa el coste de reparación de un tramo (en miles de euros).

a) ¿Puede el técnico revisar todos los nodos sin pasar dos veces por el mismo y volver al nodo inicial?

b) Si el técnico decide revisar todos los tramos de la red, ¿puede hacerlo sin pasar dos veces por el mismo tramo?

c) Si el técnico para salir del paso decide reparar sólo los tramos que permitan la conexión entre A y H. ¿Cuáles son los tramos que hay que reparar para que el coste sea mínimo?. ¿Cuál será el coste total de la reparación?.

a) Un camino hamiltoniano pasa una sola vez por cada vértice (nodo). Si el camino es cerrado se denomina ciclo hamiltoniano.

Para que hubiera un ciclo hamiltoniano las aristas incidentes en los vértices tendrían que tener grado 2.

En consecuencia, el técnico podrá revisar todos los nodos sin pasar dos veces por el mismo si la red es hamiltoniana. Para que pudiera hacerlo no tendría que pasar por los nodos K y L.

b) Un camino es euleriano cuando la trayectoria contiene todas las aristas y recorre cada arista una sola vez. Cuando el camino euleriano comienza y termina con el mismo vértice se denomina ciclo euleriano.

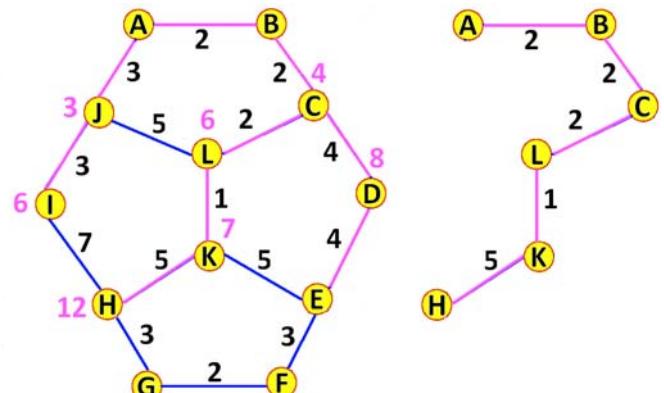
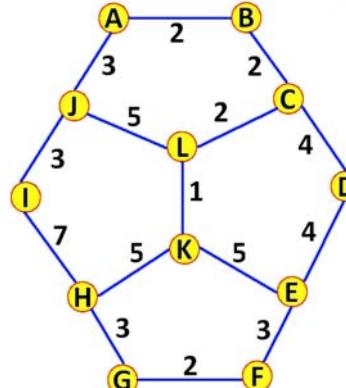
Una condición necesaria y suficiente para que la red tuviera un camino euleriano es que fuera conexa y que todos los vértices tuvieran grado par, o a lo sumo dos vértices con grado impar.

Por tanto, el técnico podría recorrer todos los tramos de la red sin pasar dos veces por el mismo tramo si admite una camino euleriano, situación que no se cumple porque hay seis nodos (C, E, H, J, L, K) con grado impar.

c) Se requiere hallar un camino de longitud o coste mínimo del nodo A al nodo H. Para ello, basta aplicar el algoritmo de Dijkstra entre los dos nodos.

Los tramos que se tienen que reparar entre los nodos A y H son los que corresponden al camino A - B - C - L - K - H que es de coste o longitud mínimo.

El coste total de la reparación sería de 12.000 euros.





Asignatura Grupo
Apellidos Nombre
Ejercicio del día

Asignatura Grupo
Apellidos Nombre
Ejercicio del día

REDES VALORADAS

- Definiciones y teoremas
- Algoritmo de Dijkstra
- Algoritmo de Ford-Fulkerson
- Flujo máximo: Programación Lineal
- Algoritmo de Prim
- Algoritmo de Kruskal
- Algoritmo de Boruvka
- Algoritmo de Sollin
- Otros Algoritmos. Aplicaciones

ÁRBOLES GENERADORES

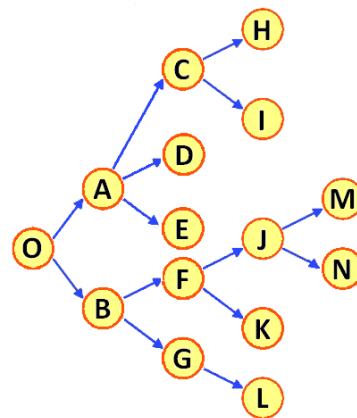
Un árbol es toda red (o grafo) $G = (V, A)$ conexa, sin ciclos y con dos vértices al menos, se designa por T.

La red adjunta representa un árbol, el vértice O recibe el nombre de *raíz del árbol*.

A los vértices o nodos se les da el nombre de hojas.

Los demás vértices reciben el nombre de *nodos de ramificación*.

Si el árbol tiene n nodos el número de aristas es $(n - 1)$.



Excluida la raíz, con los demás vértices se pueden formar subconjuntos disjuntos, cada uno de los cuales es a su vez un árbol.

Todo par de vértices está unido mediante una sola cadena.

Un grafo es un árbol \Leftrightarrow Existe un único camino simple entre todo par de vértices

Un árbol se dice binario si todos sus nodos, excepto las hojas, tienen dos sucesores inmediatos.

Un *árbol generador* de una red (o grafo) conexa y sin bucles es una red parcial de la inicial que tiene estructura de árbol.

Si una red conexa y sin bucles está valorada es un árbol mínimo (respectivamente, máximo) a un árbol generador para el cual la suma de los valores asociados a los arcos seleccionados es mínima (respectivamente, máxima)

Árbol maximal o generador de un grafo G conexo es un subconjunto maximal de G que también sea un árbol.

Un árbol maximal de G contiene todos los vértices V de G

Un conjunto de árboles recibe el nombre de bosque. Un árbol formado por un solo vértice y sin lados, se llama árbol degenerado.

Un dígrafo se denomina *árbol dirigido* cuando su grafo asociado es un árbol.

Teoremas relativos a la caracterización de Árboles:

- Dado un árbol, se verifica que $|V| = |A| + 1$
- Todas las aristas de un árbol son puentes.
- Si a un árbol se le agrega una arista entre dos de sus vértices, deja de ser un árbol.
- En un bosque de k componentes se verifica que $|V| = |A| + k$

Los problemas de redes surgen en una gran variedad de situaciones, redes de transporte, eléctricas y de comunicaciones predominan en la vida.

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

La representación de redes se utiliza en casi todos los ámbitos científicos, sociales y económicos, tal como la producción, distribución, planeación de proyectos, administración de recursos, planeación financiera, localización de instalaciones, etc.

En la actualidad se dispone de algoritmos y paquetes informáticos (WinQSB) que se utilizan de forma rutinaria para resolver problemas que no se podrían manejar hace dos décadas.

Muchos de los problemas de optimización de redes son tipos especiales de problemas de programación lineal, como pueden ser el *Problema de Transporte*, el *Problema de Asignación*, o el problema del *Flujo del Coste Mínimo*.

ÁRBOL GENERADOR MÍNIMO

Dado un grafo G conexo y etiquetado, recibe el nombre de *árbol maximal minimal o generador mínimo*, un árbol generador de G tal que la suma de los pesos de sus lados (aristas) sea mínima.

En un árbol de expansión mínima el número de aristas es la cantidad de vértices o nodos menos 1: número aristas = número de vértices – 1

Para encontrar un *árbol de expansión mínima* o *árbol recubridor* se utilizan varios algoritmos, entre otros, *algoritmo de Prim*, *algoritmo de Kruskal* y *algoritmo de Boruvka o Sollin*.

Ambos son algoritmos voraces, esto es, cada elemento a considerar se evalúa una única vez, siendo seleccionado o descartado, de tal forma que si es el elemento seleccionado forma parte de la solución, y si fuera descartado, no forma parte de la solución ni volverá a ser considerado para la misma.

El esquema del algoritmo voraz (devorador o greedy) es el que menos dificultades plantea a la hora de diseñar y comprobar su funcionamiento.

Normalmente se aplica a los problemas de optimización.

ÁRBOL GENERADOR MÁXIMO

Dado un grafo G conexo y etiquetado, recibe el nombre de *árbol maximal máximo o generador máximo*, un árbol generador de G tal que la suma de los pesos de sus lados (aristas) sea máxima.

Para encontrar un *árbol de expansión máxima* puede emplearse el algoritmo de los pesos decrecientes como si se tratase de encontrar un árbol de generador mínimo, pero con dos diferencias sobre el algoritmo anterior:

- a) Añadir aristas en lugar de suprimirlas.
- b) La conexión de una arista no deberá formar un ciclo.

En consecuencia, el *árbol maximal máximo* tiene tantos vértices como el grafo G de procedencia.

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

APLICACIONES DEL ALGORITMO DE EXPANSIÓN MÍNIMA

El algoritmo de expansión mínima se aplica en muchos diseños importantes, entre otros:

- Redes de telecomunicaciones.
- Redes de transporte para minimizar el costo total de proporcionar arcos (aristas) en carreteras, vías ferroviarias, etc.
- Cableados de equipos eléctricos.
- Red de líneas de transmisión de energía eléctrica de alto voltaje.
- Diseñar una red de tuberías para conectar varias localidades.

TERMINOLOGÍA DE REDES

Una red es un conjunto de puntos y un conjunto de líneas que unen pares de puntos.

Los puntos se llaman *nodos* (o vértices).

Las líneas se llaman *arcos* (o ligaduras, aristas o ramas). Los arcos se etiquetan al dar el nombre de los nodos en sus puntos terminales, en esta línea AB es el arco entre los nodos A y B. Los arcos de una red pueden tener un flujo de algún tipo que pase por ellos.

Cuando el flujo a través de un arco se permite solo en una dirección (como en una calle de un sentido) se denomina *arco es dirigido*, una forma de etiquetarlo es $A \rightarrow B$.

Si el flujo a través de un arco se permite en ambas direcciones se dice que es un *arco no dirigido*, con frecuencia se les denomina *ligadura*.

Una red que tiene sol arcos dirigidos se llama red dirigida, si tiene solo arcos no dirigidos se dice que es una red no dirigida. Una red con mezcla de arcos no dirigidos y dirigidos (incluso con todos sus arcos no dirigidos) se puede convertir en una red dirigida mediante la sustitución de cada arco no dirigido por un par de arcos dirigidos en direcciones opuestas. Después, según convenga, se puede proporcionar un flujo neto en una sola dirección o interpretar los flujos a través de cada par de arcos dirigidos como flujos simultáneos en direcciones opuestas.

En el proceso de toma de decisiones sobre el flujo de un *arco no dirigido* se permite hacer una secuencia de asignaciones de flujos en direcciones opuestas, entendiendo que el *flujo real* será el *flujo neto*, esto es, la diferencia de los flujos asignados en las dos direcciones.

Sea un ejemplo, se asigna un flujo de 10 en una dirección y después un flujo 4 en la dirección opuesta, el *efecto real* es la cancelación de 4 unidades de asignación original, lo que reduce el flujo en la dirección original de 10 a 6.

En *arcos dirigidos* en ocasiones se utiliza la misma técnica como una forma de reducir un flujo asignado con anterioridad.

En particular, se puede hacer una asignación ficticia de flujo en la *dirección equivocada* a través de un arco dirigido para registrar una reducción en esa cantidad del flujo que va en *dirección correcta*.

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

Se conoce como *capacidad del arco* a la cantidad máxima de flujo (quizá infinito) que puede circular en un arco dirigido.

Un *nodo fuente* (nodo de origen) tiene la propiedad de que el flujo que sale del nodo supera al que entra a él.

En un *nodo demanda* (o nodo destino) el flujo que llega excede al que sale de él.

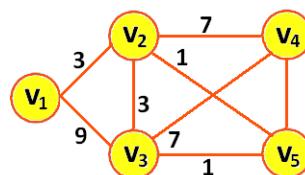
Un *nodo transbordo* (o nodo intermedio) satisface la conservación del flujo, es decir, el flujo que entra es igual al que sale.

CAMINO DE PESO MÍNIMO

En una red eléctrica o en una red vial no sólo se trata de dar servicio (conexión), es importante conocer las longitudes de las líneas para establecer cantidad de material, dificultades para su trazado (coste temporal o monetario), etc.

En el grafo adjunto a cada arista se asigna un valor que representa la magnitud (peso) que se desea considerar.

El camino 'más corto' entre el vértice v_1 y v_3 no es el que tiene menos aristas, sino aquel que recorre en total menor distancia:
 $C_1 = \{v_1, v_3\}$ recorre solo una arista con una distancia de 9.



$C_2 = \{v_1, v_2, v_3\}$ tiene dos aristas y una distancia total de 6

$C_3 = \{v_1, v_2, v_5, v_3\}$ tiene tres aristas y una distancia total de 5

El camino más corto entre los vértices v_1 y v_3 es C_3 con una distancia total de 5

El conjunto de caminos mínimos desde un vértice v a los restantes vértices de un grafo es un árbol, llamado *árbol de caminos mínimos* desde v .

Uno de los algoritmos más utilizados para la búsqueda de caminos de peso mínimo es el de Dijkstra, que proporciona los pesos mínimos desde un vértice dado al resto de los vértices.

ALGORITMO DE DIJKSTRA: CONSTRUCCIÓN DE ÁRBOL DE CAMINOS MÍNIMOS

El Algoritmo de Dijkstra, descubierto por el físico neerlandés Edsger Dijkstra en 1959, resuelve el problema de hallar el camino de longitud mínima entre dos vértices de un grafo ponderado.

El algoritmo va explorando los caminos más cortos que parten del vértice origen y que llevan a todos los demás vértices, el algoritmo finaliza al obtener el camino más corto que lleva a todos los vértices.

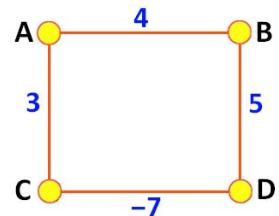
El algoritmo de Dijkstra devuelve en realidad el peso mínimo, no el camino mínimo propiamente dicho, pero permite obtener fácilmente el camino mínimo recorriendo en *sentido inverso de la construcción*.

Es un algoritmo de búsqueda de peso uniforme y, en consecuencia no funciona en grafos con aristas de peso negativo. Al elegir el nodo (vértice) con menor distancia quedan excluidos de la búsqueda nodos que en sucesivas iteraciones bajarían el peso general del camino al pasar por una arista con peso negativo.

El algoritmo de Dijkstra puede producir respuestas incorrectas si tiene pesos negativos.

La ruta más corta entre A y C no es 3, en realidad es A → B → D → C

Se recurre al algoritmo de Bellman-Ford



Con el algoritmo de Dijkstra se pueden resolver grafos con muchos vértices, que sería muy complicado resolver con otros algoritmos, por lo que tiene una aplicación importante en tecnología y comunicaciones.

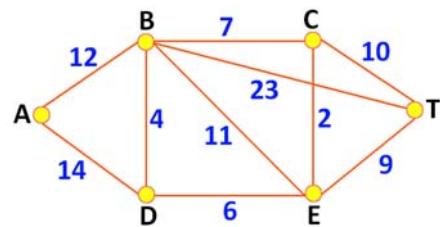
El conjunto de caminos mínimos siempre es un árbol generador.

El algoritmo de Dijkstra devuelve el peso mínimo no el camino mínimo. Por lo que en general, el árbol obtenido no tiene por qué ser mínimo.

A continuación se resuelve el problema de encontrar el camino de longitud mínima entre dos vértices de un grafo ponderado no dirigido si todos los pesos no son negativos. El Algoritmo puede adaptarse para resolver problemas de longitud mínima entre grafos dirigidos.

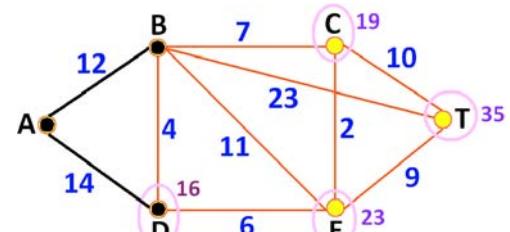
Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

- DIJKSTRA: Hallar los caminos de longitud mínima desde A hasta cada uno de los restantes vértices del grafo.



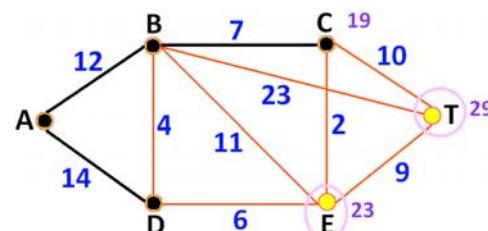
Se eligen los nodos adyacentes que tienen menor peso en la arista. Se elige el nodo B , marcando la arista AB.

Surgen candidatos, el vértice D es el nodo que con 14 tiene menor peso en la arista. Se marca la arista AD.

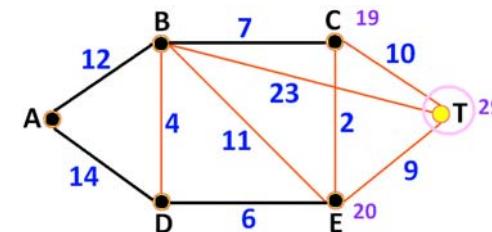


Se marca la arista BC porque es el nodo que con 19 tiene menor peso en la arista.

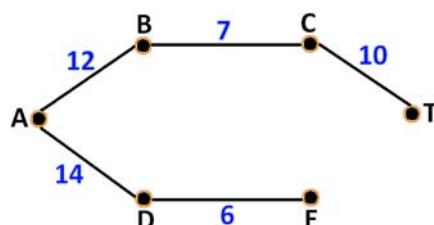
Cambia el peso de la arista CT a 29.



Se marca la arista DE con peso 20



El árbol de longitud mínima se muestra en la figura adjunta.

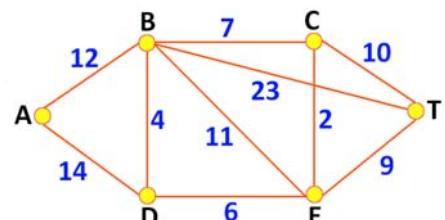


ELABORACIÓN DE LA TABLA DE DIJKSTRA

El algoritmo de Dijkstra construye un árbol de caminos mínimos desde el vértice A hasta los restantes vértices del grafo.

El algoritmo atribuye a cada vértice v una etiqueta $t(v)$ que es la longitud del camino ya seleccionado.

Se inicia la tabla con $t(A) = 0$, $t(v) = \infty \quad \forall v \neq A$



A	B	C	D	E	T	Vértice	Arista
0°	∞	∞	∞	∞	∞	A	

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

El vértice es el más pequeño del árbol que se va construyendo, se marca y se incorpora al árbol.

Se toman todas las aristas Av , con el resto de vértices v no pertenecientes al árbol y se actualizan las etiquetas: $\min[t(v), t(A) + p(\overline{Av})]$.

Aparece una nueva fila en la tabla:

A	B	C	D	E	T	Vértice	Arista
0°	∞	∞	∞	∞	∞	A	
	12°	∞	14	∞	∞	B	AB

El vértice con menor etiqueta, en este caso B se marca, se añade al árbol de caminos mínimos y se incorpora la arista AB.

Se toman todas las aristas Bv , con el resto de vértices v no pertenecientes al árbol y se actualizan las etiquetas de los vértices adyacentes de B, como antes, $\min[t(v), t(B) + p(\overline{Bv})]$, resultando:

$$t(D) = 14 < t(B) + p(\overline{BD}) = 12 + 4 = 16 \quad \text{No cambia}$$

$$t(C) = \infty > t(C) + p(\overline{BC}) = 12 + 7 = 19 \quad \text{Cambia}$$

$$t(E) = \infty > t(E) + p(\overline{BE}) = 12 + 11 = 23 \quad \text{Cambia}$$

$$t(T) = \infty > t(T) + p(\overline{BT}) = 12 + 23 = 35 \quad \text{Cambia}$$

En la tabla aparece una nueva fila con las etiquetas, añadiendo la etiqueta mínima D y la arista AD.

A	B	C	D	E	T	Vértice	Arista
0°	∞	∞	∞	∞	∞	A	
	12°	∞	14	∞	∞	B	AB
		19	14°	23	35	D	AD

Ahora se observan los vértices adyacentes a D que no están en el árbol y se actualizan las etiquetas.

Sólo hay el vértice vecino E: $\min[t(E), t(D) + p(\overline{DE})]$

$$t(E) = 23 > t(D) + p(\overline{DE}) = 14 + 6 = 20 \quad \text{Cambia}$$

A	B	C	D	E	T	Vértice	Arista
0°	∞	∞	∞	∞	∞	A	
	12°	∞	14	∞	∞	B	AB
		19	14°	23	35	D	AD
		19°		20	35	C	BC

- De otra parte, para saber cuál es el vértice vecino a C en el árbol, se sube por la columna de C hasta encontrar la primera variación en las etiquetas.

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

El vértice marcado en esa fila será el vecino.

La primera variación que se encuentra es el salto de 19 a ∞ , que se produce en la fila donde está 12° , en consecuencia, es el vértice B

A	B	C	D	E	T	Vértice	Arista
0°	∞	∞	∞	∞	∞	A	
	12°	∞	14	∞	∞	B	\overline{AB}
		19	14°	23	35	D	\overline{AD}
		19°		20	35	C	\overline{BC}

Los vértices adyacentes al vértice C son los vértices E y T.

Se actualizan las etiquetas de los vértices adyacentes de C: $\min[t(v), t(C) + p(Cv)]$, resultando:

$$t(E) = 20 < t(C) + p(CE) = 19 + 2 = 21 \quad \text{No cambia}$$

$$t(T) = 35 > t(C) + p(CT) = 19 + 10 = 29 \quad \text{Cambia}$$

A	B	C	D	E	T	Vértice	Arista
0°	∞	∞	∞	∞	∞	A	
	12°	∞	14	∞	∞	B	\overline{AB}
		19	14°	23	35	D	\overline{AD}
		19°		20	35	C	\overline{BC}
				20°	29	E	\overline{DE}

El vértice con etiqueta mínima es E, se añade al árbol el vértice E y la arista \overline{DE} .

El vértice vecino a E en el árbol se encuentra al subir por la columna E hasta encontrar la primera variación en las etiquetas, el vértice marcado es 14° , con lo que es el vértice D, la ruta \overline{DE} tiene una longitud $20 - 14 = 6$

Continúa el algoritmo, desde el vértice E sólo queda un vértice adyacente, T. Se actualiza la etiqueta: $\min[t(T), t(E) + p(ET)]$

$$t(T) = 29 = t(E) + p(ET) = 20 + 9 = 29 \quad \text{No Cambia}$$

A	B	C	D	E	T	Vértice	Arista
0°	∞	∞	∞	∞	∞	A	
	12°	∞	14	∞	∞	B	\overline{AB}
		19	14°	23	35	D	\overline{AD}
		19°		20	35	C	\overline{BC}
				20°	29	E	\overline{DE}
					29°	T	\overline{CT}

El vértice vecino a T en el árbol se encuentra al subir por la columna T hasta encontrar la primera variación en la etiqueta, se encuentra en el vértice C.

La ruta CT tiene una longitud $29 - 19 = 10$

Se añade al árbol el vértice T y la arista CT.

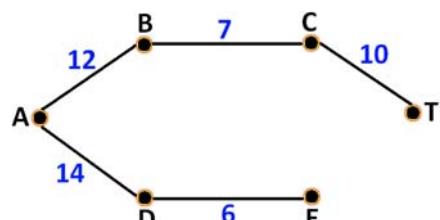
- Para hallar las rutas más cortas desde el vértice A a los restantes vértices no es necesario representar el grafo, basta con los datos de la tabla:

A	B	C	D	E	T
0°	∞	∞	∞	∞	∞
	12°	∞	14	∞	∞
		19	14°	23	35
			19°	20	35
				20°	29
					29°

Vecino B el vértice A: Ruta AB. Longitud 12

Observando los vértices vecinos:

Vecino C el vértice B: Ruta BC. Longitud 7



Vecino D el vértice A: Ruta AD. Longitud 14

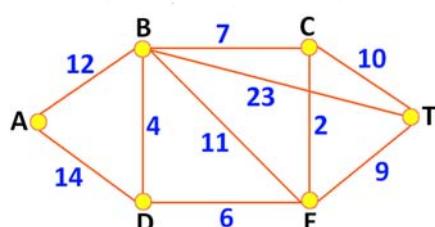
Vecino E el vértice D: Ruta DE. Longitud 6

Vecino T el vértice C: Ruta CT. Longitud 10



RTA MÁS CORTA: [WinQSB / Network Modeling - Shortest Path Problem](#)

Encontrar la ruta más corta desde el vértice A hasta cada uno de los restantes vértices del grafo.



Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

NET Problem Specification

Problem Type	Objective Criterion	
<input type="radio"/> Network Flow <input type="radio"/> Transportation Problem <input type="radio"/> Assignment Problem <input checked="" type="radio"/> Shortest Path Problem <input type="radio"/> Maximal Flow Problem <input type="radio"/> Minimal Spanning Tree <input type="radio"/> Traveling Salesman Problem	<input checked="" type="radio"/> Minimization <input type="radio"/> Maximization	
Data Entry Format		
<input checked="" type="radio"/> Spreadsheet Matrix Form <input type="radio"/> Graphic Model Form <input type="checkbox"/> Symmetric Arc Coefficients <i>(i.e., both ways same cost)</i>		
Problem Title	DIJKSTRA	
Number of Nodes	6	
OK	Cancel	Help

Network Modeling

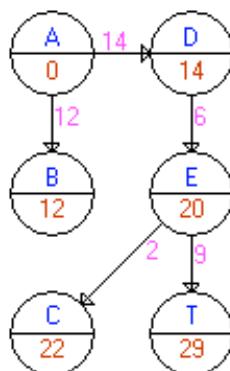
File Edit Format Solve and Analyze Results Utilities Window WinQSB Help

Shortest Path Problem DIJKSTRA: RUTA MÁS CORTA

From \ To	A	B	C	D	E	T
A		12		14		
B	12			4	11	23
C		7			2	10
D	14	4			6	
E			2	6		9
T		23	10		9	

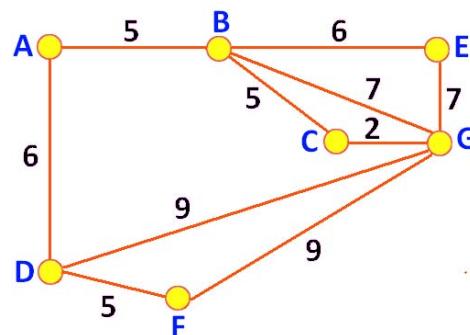
Solution for Shortest Path Problem DIJKSTRA: RUTA MÁS CORTA

	From	To	Distance/Cost	Cumulative Distance/Cost
1	A	D	14	14
2	D	E	6	20
3	E	T	9	29
	From A	To T	=	29
	From A	To B	=	12
	From A	To C	=	22
	From A	To D	=	14
	From A	To E	=	20



DIJKSTRA: Hallar la ruta más corta desde el vértice A hasta cada uno de los restantes vértices del grafo ponderado. ¿El conjunto de caminos construidos constituye un árbol generador del grafo?

	B	C	D	E	F	G
A	5		6			
B		5		6		7
C						2
D					5	9
E						7
F						9

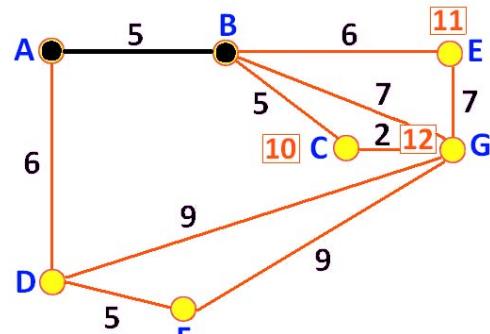


Grafo del planteamiento:

Se eligen los nodos adyacentes que tienen menor peso en la arista.

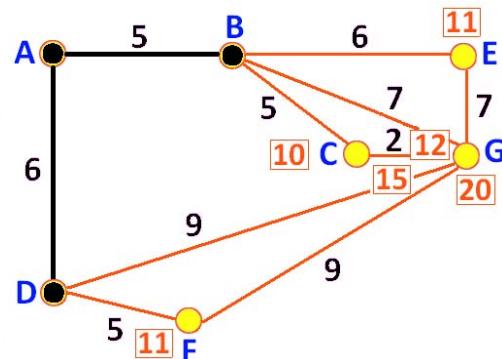
Se elige el nodo B

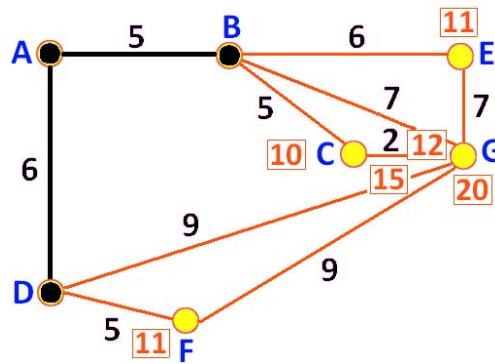
Vértice B ruta AB Longitud: 5



Se elige el nodo D

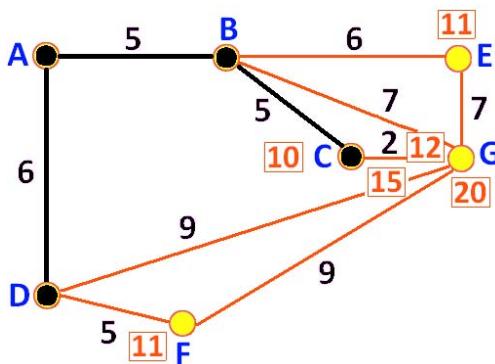
Vértice D ruta AD Longitud: 6





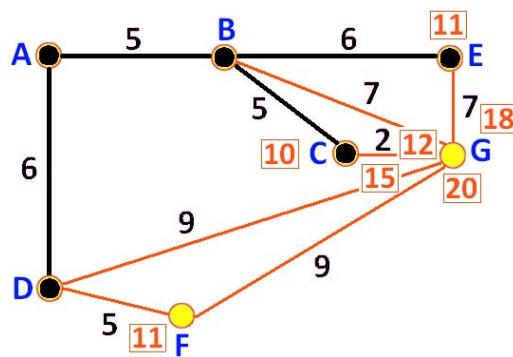
Se elige el nodo D

Vértice D ruta AD Longitud: 6



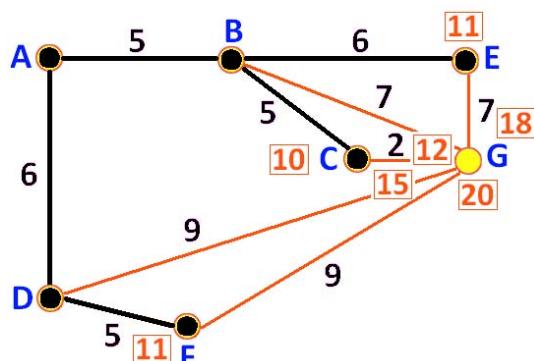
Se elige el nodo C

Vértice C ruta ABC Longitud: 10



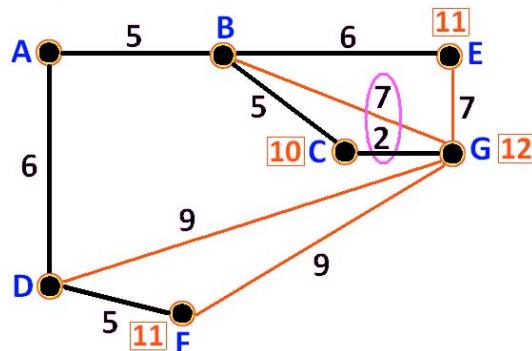
Se elige el nodo E

Vértice E ruta ABE Longitud: 11



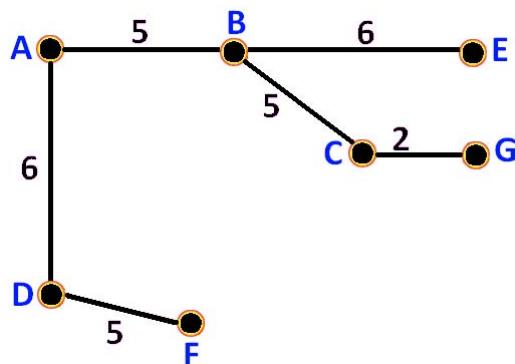
Se elige el nodo F

Vértice F ruta ADF Longitud: 11



Se elige el nodo G

Vértice F ruta ABCG Longitud: 12



El árbol de mínimos caminos desde A es:

El conjunto de caminos mínimos siempre es un árbol generador.

El algoritmo de Dijkstra devuelve el peso mínimo no el camino mínimo. Por lo que en general, el árbol obtenido no tiene por qué ser mínimo.

En este caso, sí es un árbol generador.

- No es necesario representar el grafo. Se puede elaborar la TABLA DE DIJKSTRA

	B	C	D	E	F	G
A	5		6			
B		5		6		7
C						2
D					5	9
E						7
F						9

El algoritmo atribuye a cada vértice v una etiqueta $t(v)$, que es la longitud del camino ya seleccionado. Se inicia la tabla con $t(A) = 0$, $t(v) = \infty \quad \forall v \neq A$

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

Tiene dos vértices vecinos B y C que no cambian. Se elige el vértice con menor etiqueta, que es el B con un peso de 5. Se añade al árbol y se marca la arista \overline{AB} que se incorpora al árbol de caminos mínimos.

A	B	C	D	E	F	G	Vértice	Arista
	∞	∞	∞	∞	∞	∞	A	
	5*	∞	6	∞	∞	∞	B	\overline{AB}
		10	6*	11	∞	12	D	\overline{AD}
			10*		11	11	C	\overline{BC}
				11*	11	12	E	\overline{BE}
					11*	12	F	\overline{DF}
						12*	G	\overline{CG}

- Aparece una nueva fila: Se marca el vértice D con menor etiqueta (6) y se marca la arista \overline{AD} , que se incorpora al árbol de caminos mínimos.

Se toman todas las aristas \overline{Bv} , con el resto de vértices v no pertenecientes al árbol y se actualizan etiquetas de los vértices adyacentes a B: $\min[t(v), t(B) + p(\overline{Bv})]$

$$t(C) = \infty > t(B) + p(\overline{BC}) = 5 + 5 = 10 \quad \text{Cambia}$$

$$t(E) = \infty > t(B) + p(\overline{BE}) = 5 + 6 = 11 \quad \text{Cambia}$$

$$t(G) = \infty > t(B) + p(\overline{BG}) = 5 + 7 = 12 \quad \text{Cambia}$$

- Aparece una nueva fila: Se marca el vértice C que tiene menor etiqueta (10) y se marca la arista \overline{BC} , que se incorpora al árbol.

Se actualizan las etiquetas de los vértices adyacentes a D: $\min[t(v), t(D) + p(\overline{Dv})]$

$$t(F) = \infty > t(D) + p(\overline{DF}) = 6 + 5 = 11 \quad \text{Cambia}$$

$$t(G) = 12 < t(D) + p(\overline{DG}) = 6 + 9 = 15 \quad \text{No Cambia}$$

- Aparece una nueva fila: Se marca el vértice E que tiene menor etiqueta (11) y se marca la arista \overline{BE} y se incorpora al árbol.

Se actualiza la etiqueta del vértice adyacente a C: $\min[t(v), t(C) + p(\overline{Cv})]$

$$t(G) = 12 = t(C) + p(\overline{CG}) = 10 + 2 = 12 \quad \text{No Cambia}$$

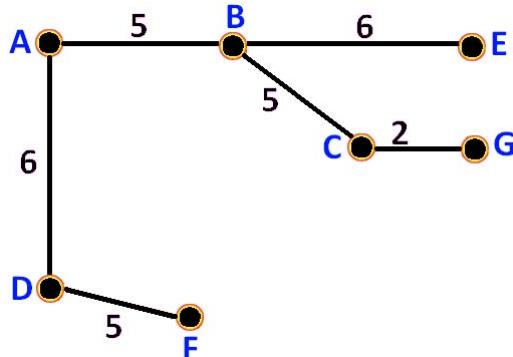
- Aparece una nueva fila: Se marca el vértice E, incorporando la arista \overline{DF} al árbol.

Se actualiza la etiqueta del vértice adyacente a E: $\min[t(v), t(E) + p(\overline{Ev})]$

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

$$t(G) = 12 < t(E) + p(\overline{EG}) = 11 + 7 = 18 \text{ No Cambia}$$

Vértice	Camino	Longitud
B	AB	5
D	AD	6
C	ABC	10
E	ABE	11
F	ADF	11
G	ABCG	12



Max. **CX** **Ax=b** DIJKSTRA - RUTA MÁS CORTA: WinQSB / Network Modeling - Shortest Path Problem

(IMPRESO EN PAPEL RECICLADO)

UNICAMENTE PARA USO ESCOLAR

NET Problem Specification

Problem Type

- Network Flow
- Transportation Problem
- Assignment Problem
- Shortest Path Problem
- Maximal Flow Problem
- Minimal Spanning Tree
- Traveling Salesman Problem

Objective Criterion

- Minimization
- Maximization

Data Entry Format

- Spreadsheet Matrix Form
- Graphic Model Form
- Symmetric Arc Coefficients (i.e., both ways same cost)

Problem Title DIJKSTRA

Number of Nodes 7

OK **Cancel** **Help**

Network Modeling

File Edit Format Solve and Analyze Results Utilities Window WinQSB Help

Shortest Path Problem DIJKSTRA2

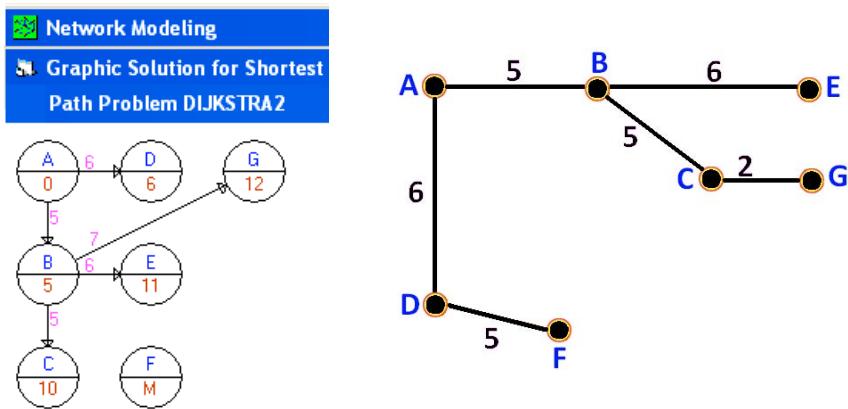
From \ To	A	B	C	D	E	F	G
A		5		6			
B	5		5		6		7
C		5					2
D	6						9
E		6					7
F				6			9
G		7	2		7		

Network Modeling

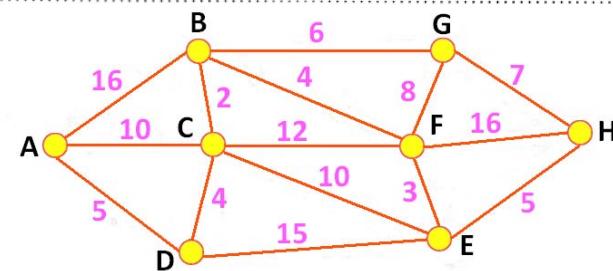
File Format Results Utilities Window Help

0.00 A

	From	To	Distance/Cost	Cumulative Distance/Cost
1	A	B	5	5
2	B	G	7	12
	From A	To G	=	12
	From A	To B	=	5
	From A	To C	=	10
	From A	To D	=	6
	From A	To E	=	11
	From A	To F	=	M



- DIJKSTRA: Encontrar el camino de longitud mínima de unas estaciones de metro entre A y H.

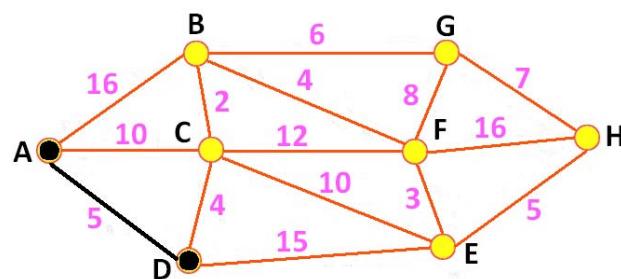


Se eligen los nodos adyacentes que tienen menor peso en la arista.

Vértice D

Ruta AD

Longitud: 5

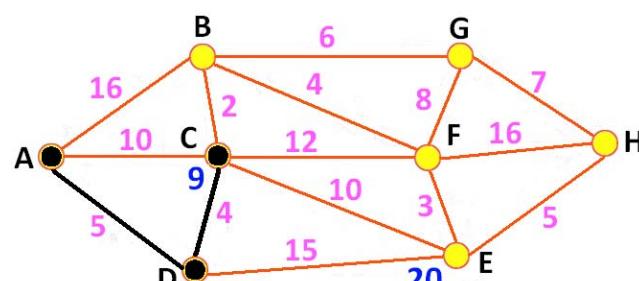


Surgen candidatos, vértices B, C y E.
El camino mínimo resulta el vértice C.

Vértice C

Ruta ADC

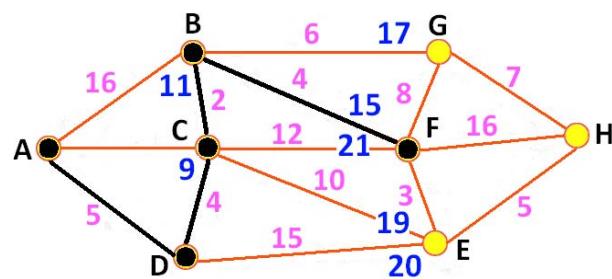
Longitud: 9



Vértice B

Ruta ADCB

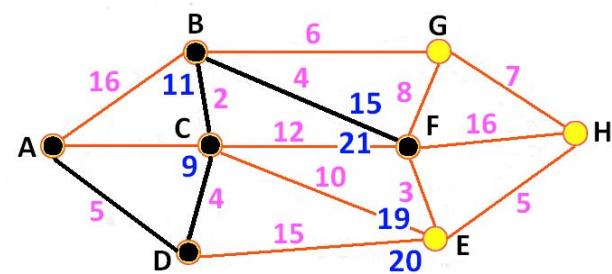
Longitud: 11



Vértice F

Ruta ADCBF

Longitud: 15

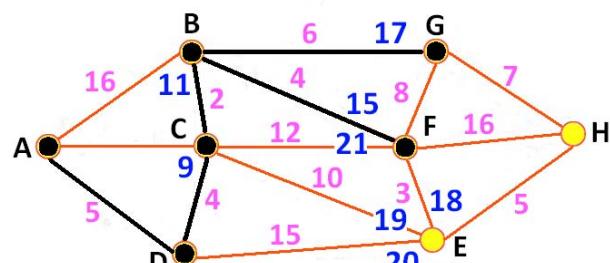


Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

Vértice G

Ruta ADCBG

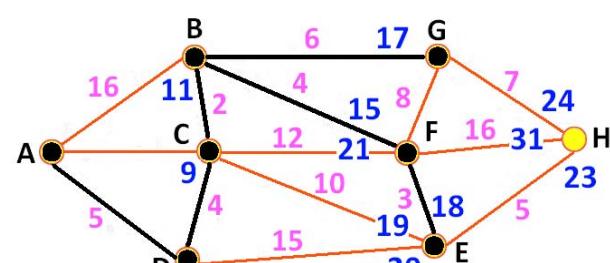
Longitud: 17



Vértice E

Ruta ADCBFE

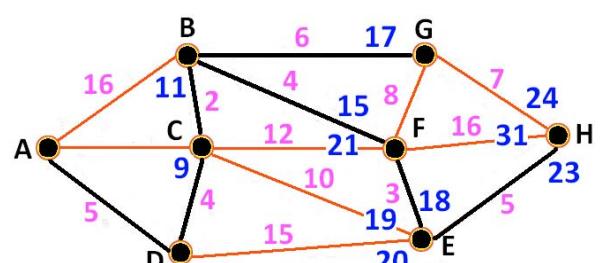
Longitud: 18



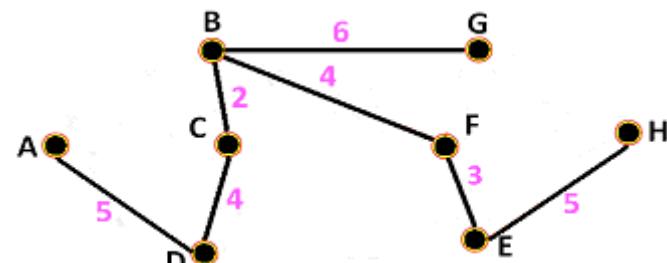
Vértice H

Ruta ADCBFEH

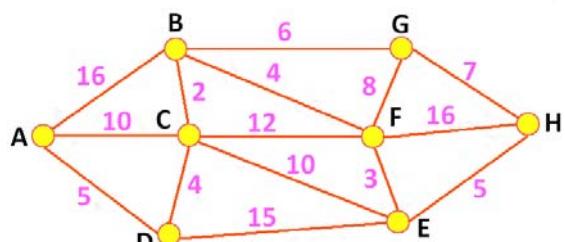
Longitud: 23



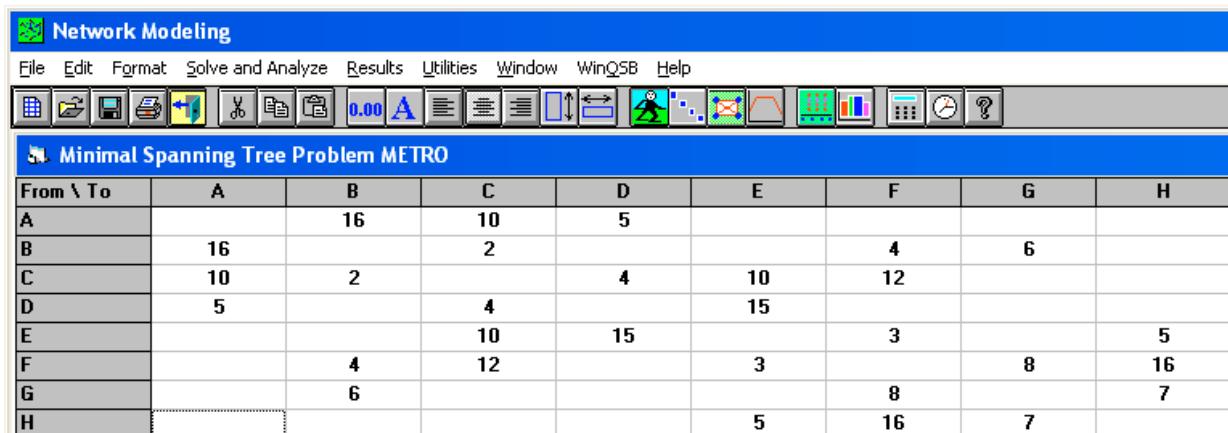
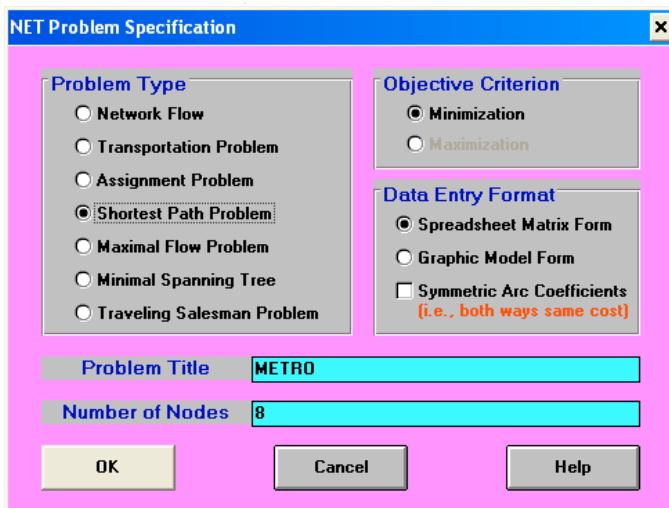
El árbol de caminos mínimos es:

RUTA MÁS CORTA: [WinQSB / Network Modeling - Shortest Path Problem](#)

Encontrar el camino de longitud mínima
de unas estaciones de metro entre A y H.

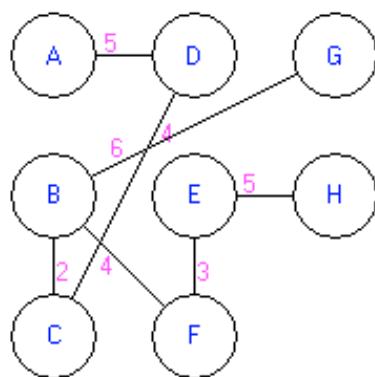


Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día



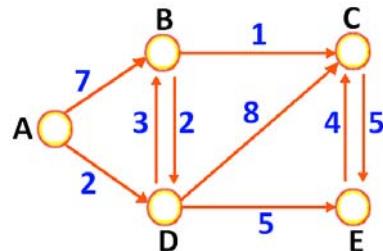
Solution for Minimal Spanning Tree Problem METRO

	From Node	Connect To	Distance/Cost		From Node	Connect To	Distance/Cost
1	C	B	2	5	B	F	4
2	D	C	4	6	B	G	6
3	A	D	5	7	E	H	5
4	F	E	3				
	Total	Minimal	Connected	Distance or Cost	=		29

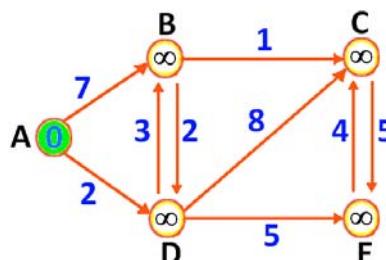


ALGORITMO DE DIJKSTRA EN GRAFOS DIRIGIDOS

- Hallar la ruta más corta desde el vértice A hasta cada uno de los restantes vértices del grafo dirigido.

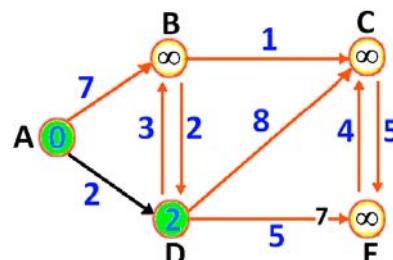


El algoritmo de Dijkstra se inicia partiendo del vértice A.



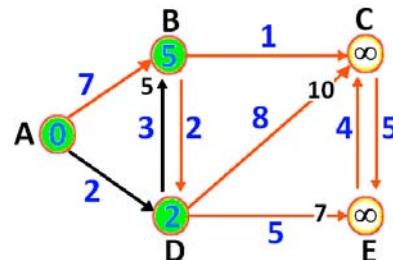
Los vértices adyacentes al vértice A son B y D, se elige el vértice D por tener menor peso.

Vértice D ruta AD Peso: 2



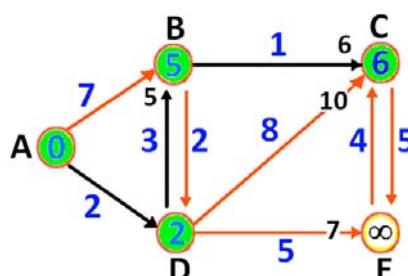
Entre los posibles vértices no visitados, se elige B por qué con un peso de 5 es el más pequeño.

Vértice B ruta ADB Peso: 5



Entre los posibles vértices se elige C, con peso 6 es el más pequeño.

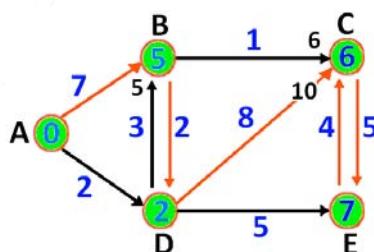
Vértice C ruta ADCB Peso: 6



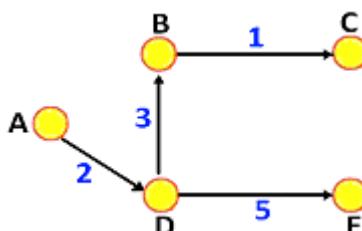
Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

Finalmente,
se elige E con un peso de 7

Vértice E ruta ADE Peso: 7

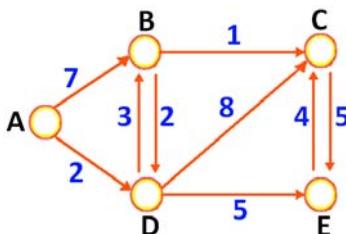


El árbol de caminos mínimos
desde A es:



WinQSB / Network Modeling - Shortest Path Problem

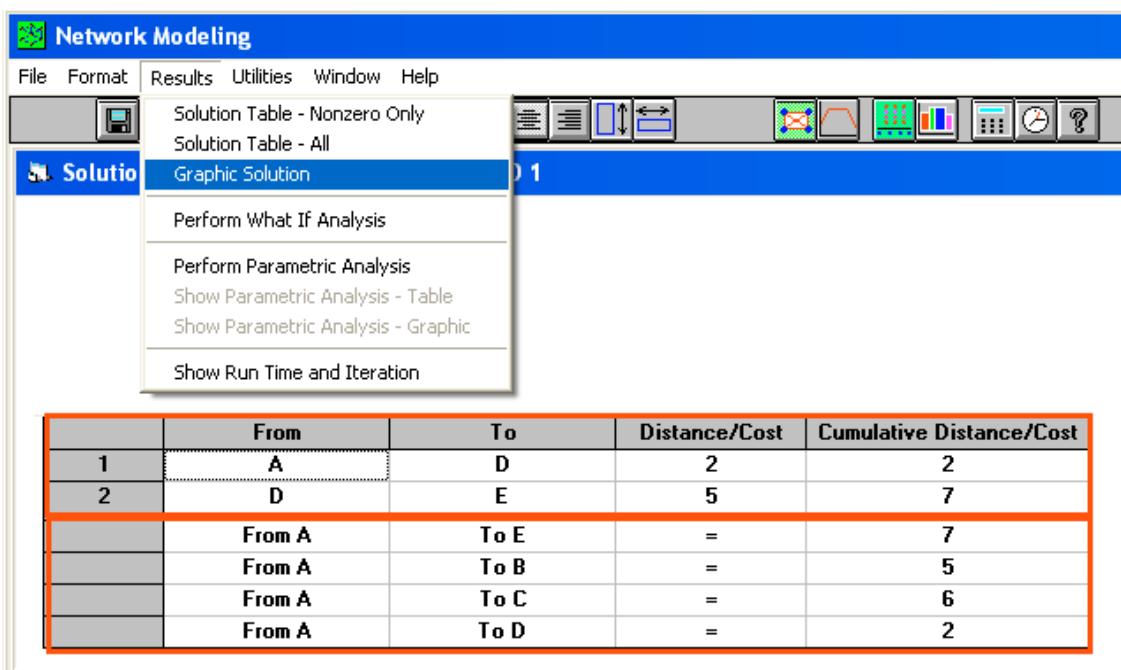
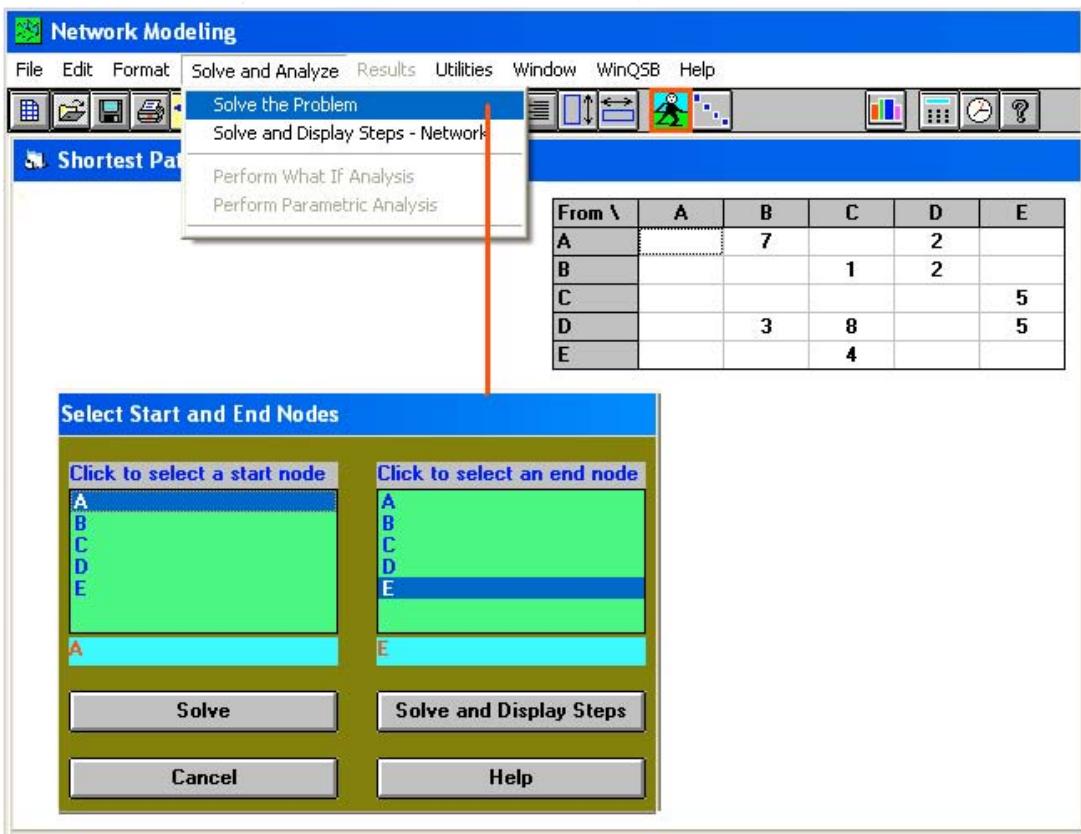
Hallar la ruta más corta desde el vértice A hasta cada uno de los restantes vértices del grafo dirigido.



NET Problem Specification

Problem Type	Objective Criterion	
<input type="radio"/> Network Flow <input type="radio"/> Transportation Problem <input type="radio"/> Assignment Problem <input checked="" type="radio"/> Shortest Path Problem <input type="radio"/> Maximal Flow Problem <input type="radio"/> Minimal Spanning Tree <input type="radio"/> Traveling Salesman Problem	<input checked="" type="radio"/> Minimization <input type="radio"/> Maximization	
Data Entry Format		
<input checked="" type="radio"/> Spreadsheet Matrix Form <input type="radio"/> Graphic Model Form <input type="checkbox"/> Symmetric Arc Coefficients <small>[i.e., both ways same cost]</small>		
Problem Title	GRAFO 1	
Number of Nodes	5	
OK	Cancel	Help

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día



Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

DIJKSTRA: Una empresa de alquiler de automóviles está estudiando el plan de renovación de su flota de coches para los próximos siete semestres (2020-2023) de julio de 2020 a finales de 2023.

Al principio de cada semestre, enero a julio, se toma la decisión acerca de si debe mantener durante ese semestre el coche o por el contrario si se debe remplazar por otro nuevo.

Suponiendo que cada coche se debe mantener al menos un año (dos semestres) y a lo sumo dos años (cuatro semestres), determinar la política de reemplazo para ese período con el mínimo coste, considerando los datos de la tabla adjunta que, muestra el coste estimado de reemplazamiento (en euros) en función del semestre en el cual se adquiere y el número de semestres en servicio, para un modelo determinado.

	Dos semestres	Tres semestres	Cuatro semestres
Segundo semestre 2020	3.800	4.100	6.800
Primer semestre 2021	4.000	4.800	7.000
Segundo semestre 2021	4.200	5.100	7.200
Primer semestre 2022	4.800	5.700	-----
Segundo semestre 2022	5.700	-----	-----

A = "segundo semestre 2020"

D = "primer semestre 2022"

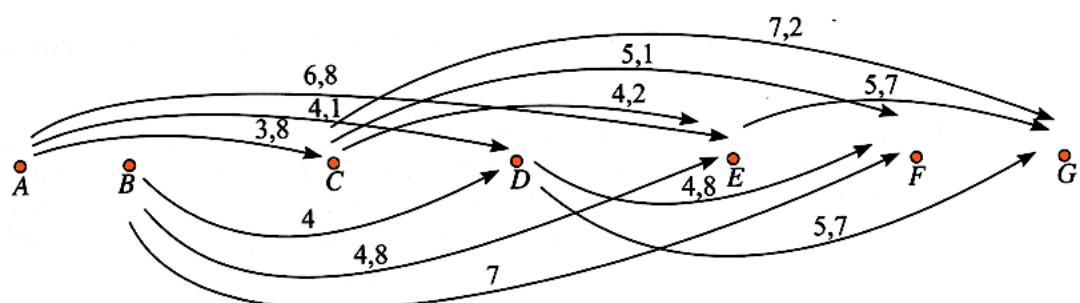
Denotando los sucesos: B = "primer semestre 2021"

E = "segundo semestre 2022"

C = "segundo semestre 2021"

F = "primer semestre 2023"

Una red que representa la situación descrita sería:



Aplicando el algoritmo de Dijkstra para hallar el camino mínimo desde el vértice A a todos los demás, excluido B que no conecta con A

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

A	C	D	E	F	G
0*	3,8	4,1	6,8	∞	∞
	↓	↓	↓	↓	↓
3,8*	4,1	6,8	8,9	11	
	↓	↓	↓	↓	
4,1*	6,8	8,9	9,8		
	↓	↓	↓		
6,8*	8,9	9,8			$\Leftarrow G: \min\{9,8; 4,1+5,7\} = 9,8$
	↓	↓			
8,9*	9,8				$\Leftarrow G: \min\{9,8; 6,8+5,7\} = 9,8$
	↓				
9,8*					

El camino mínimo es: A $\xrightarrow{4,1}$ D $\xrightarrow{5,7}$ G \rightarrow Coste mínimo 9.800 euros

Por tanto, el coche comprado en el segundo semestre del año 2020, se cambia en el primer semestre del año 2022 y se mantiene el segundo semestre del año 2023.

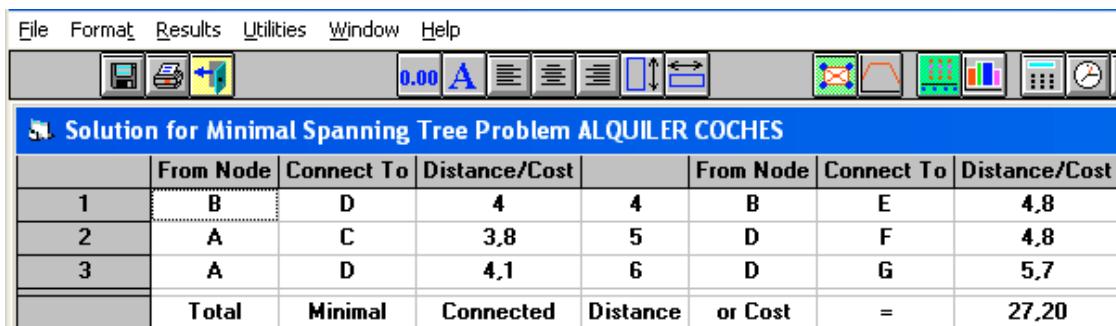
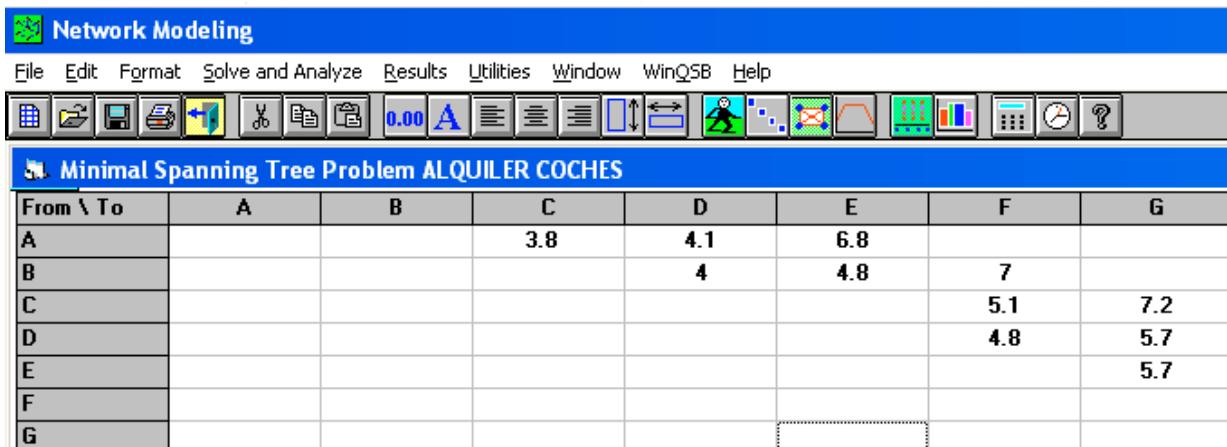


WinQSB / Network Modeling - Shortest Path Problem

NET Problem Specification

Problem Type	Objective Criterion	
<input type="radio"/> Network Flow <input type="radio"/> Transportation Problem <input type="radio"/> Assignment Problem <input checked="" type="radio"/> Shortest Path Problem <input type="radio"/> Maximal Flow Problem <input type="radio"/> Minimal Spanning Tree <input type="radio"/> Traveling Salesman Problem	<input checked="" type="radio"/> Minimization <input type="radio"/> Maximization	
Data Entry Format		
<input checked="" type="radio"/> Spreadsheet Matrix Form <input type="radio"/> Graphic Model Form <input type="checkbox"/> Symmetric Arc Coefficients <small>[i.e., both ways same cost]</small>		
Problem Title	ALQUILER COCHES	
Number of Nodes	7	
OK	Cancel	Help

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día



Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

FLUJO MÁXIMO EN REDES

En teoría de grafos, un grafo dirigido con pesos es también conocido como una red.

Se determina el Flujo Máximo porque hay innumerables cuestiones prácticas donde lo más importante es conocer la cantidad de flujo que pasa a través de una red.

El problema del Flujo Máximo consiste: Dado un grafo dirigido con pesos, $G = (V, A, W)$, que representa las capacidades máximas de los canales, un nodo de inicio S y otro de fin T en V , se trata de encontrar la cantidad máxima de flujo que puede circular desde S hasta T .

Las aristas representan canales por los que puede circular cierta cosa: transmisión de datos, redes de corriente eléctrica, líneas de oleoductos, agua, automóviles, etc.

Los pesos de las aristas representan la capacidad máxima de un canal: velocidad de una conexión, cantidad máxima de tráfico, voltaje de una línea eléctrica, volumen máximo de agua, etc.

Los problemas de Flujo Máximo se pueden resolver mediante programas informáticos, por ejemplo, el programa WinQSB con un conjunto de herramientas útiles para la investigación de operaciones. Dentro de WinQSB se encuentra el modulo Network Modeling (Maximal Flow Problem), que permite resolver problemas de Flujo Máximo con facilidad.

CONCEPTOS BÁSICOS:

- Flujo: Circulación de unidades homogéneas de un lugar a otro.
- Capacidad de flujo: Capacidad de unidades que pueden entrar por el nodo fuente y salir por el nodo destino.
- Origen o fuente de flujo: Nodo por el que ingresa el flujo.
- Destino o Sumidero de flujo: Nodo por que sale el flujo.
- Capacidades residuales: Capacidades restantes una vez que el flujo pasa el arco.

MÉTODO DE FORD-FULKERSON: FLUJO MÁXIMO EN REDES

El método propone buscar caminos en los que se pueda aumentar el flujo hasta que se alcance el flujo máximo, la idea es encontrar una ruta de penetración con un flujo positivo neto que une los nodos de origen y destino.

- El flujo es siempre positivo y con unidades enteras.
- El flujo a través de un arco es menor o igual que la capacidad.
- El flujo que entra en un nodo es igual al que sale de él.

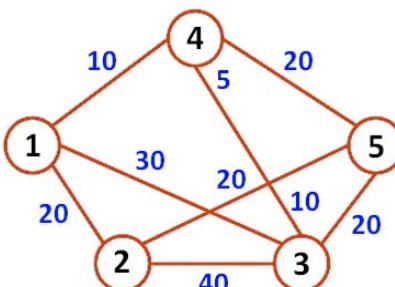
PASOS PARA LA RESOLUCIÓN DE UN PROBLEMA DE FLUJO MÁXIMO

1. Identificar el nodo origen y de destino.
2. Partiendo del nodo de origen se elige el arco que posea mayor flujo
3. Identificar los nodos de transbordo.
4. Repetir el proceso como si el nodo intermedio fuera el nodo origen.
5. Calcular 'k' y las nuevas capacidades.
6. Obtenido el resultado se cambian las capacidades y se repite idéntico procedimiento desde el inicio.

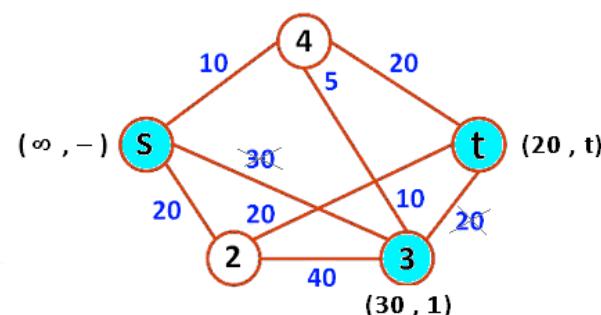
$$C_{ij, ji} = (C_i - k, C_j + k) \quad \left\{ \begin{array}{l} C \equiv \text{Capacidad} \\ ij \equiv \text{Índices de los nodos} \\ k \equiv \text{Mínimo flujo que pasa por el nodo} \\ k = \min(\text{capacidades de la ruta}) \end{array} \right.$$

El Flujo Máximo que puede pasar del nodo origen hasta el nodo destino es la suma de las capacidades $\sum k$ de la ruta.

Calcular el flujo máximo del grafo:



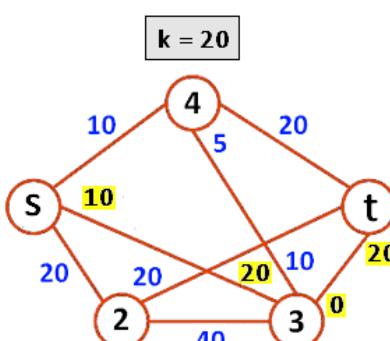
MÉTODO DE FORD-FULKERSON: Flujo máximo desde s, remplazando nuevas capacidades



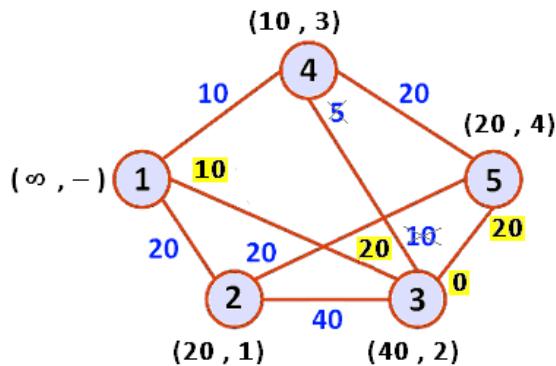
$$k = \min(\infty, 30, 20) = 20$$

$$C_{13,31} = (30 - 20, 0 + 20) = (10, 20)$$

$$C_{35,53} = (20 - 20, 0 + 20) = (0, 20)$$



Ruta: 1 - 2 - 3 - 4 - 5 / Remplazando nuevas capacidades



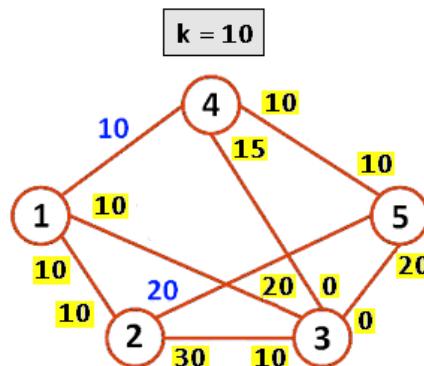
$$k = \min(\infty, 20, 40, 10, 20) = 10$$

$$C_{12,21} = (20 - 10, 0 + 10) = (10, 10)$$

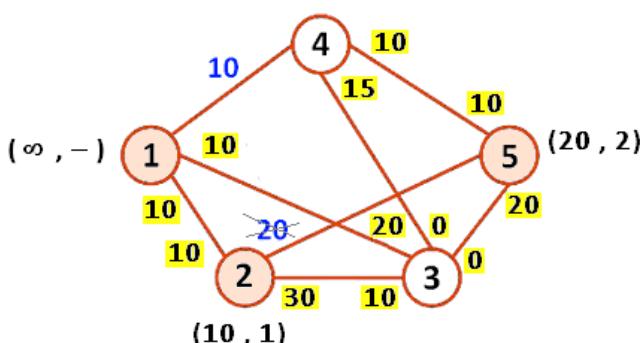
$$C_{23,32} = (40 - 10, 0 + 10) = (30, 10)$$

$$C_{34,43} = (10 - 10, 5 + 10) = (0, 15)$$

$$C_{45,54} = (20 - 10, 0 + 10) = (10, 10)$$



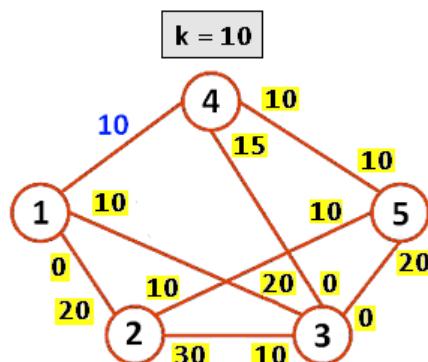
Ruta: 1 - 2 - 5 / Remplazando nuevas capacidades



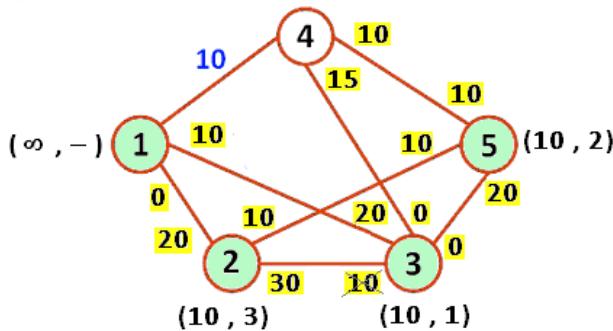
$$k = \min(\infty, 10, 20) = 10$$

$$C_{12,21} = (10 - 10, 10 + 10) = (0, 20)$$

$$C_{25,52} = (20 - 10, 0 + 10) = (10, 10)$$



Ruta: 1 - 3 - 2 - 5 / Remplazando nuevas capacidades

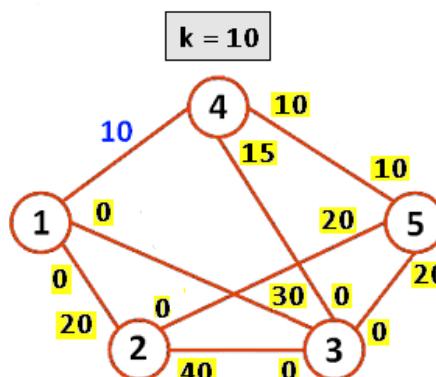


$$k = \min(\infty, 10, 10, 10) = 10$$

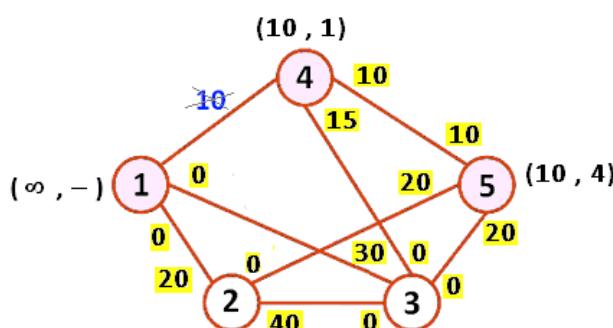
$$C_{13,31} = (10 - 10, 20 + 10) = (0, 30)$$

$$C_{32,23} = (10 - 10, 30 + 10) = (0, 40)$$

$$C_{25,52} = (10 - 10, 10 + 10) = (0, 20)$$



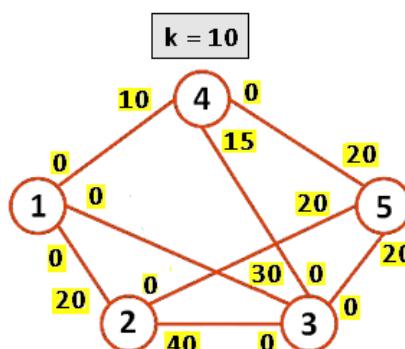
Finalmente, Ruta: 1 - 4 - 5 / Remplazando nuevas capacidades



$$k = \min(\infty, 10, 10) = 10$$

$$C_{14,41} = (10 - 10, 0 + 10) = (0, 10)$$

$$C_{45,54} = (10 - 10, 10 + 10) = (0, 20)$$



El Flujo máximo se obtiene al sumar todas las nuevas capacidades:

$$\sum k = 20 + 10 + 10 + 10 + 10 = 60$$

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día



Network Modeling / Maximal Flow Problem

Muchos problemas pueden ser modelados mediante una red en donde los arcos limitan la cantidad de un producto que se puede enviar. En estas situaciones, frecuentemente se transporta la máxima cantidad de flujo desde un punto de partida llamado fuente hacia un punto final denominado pozo.

NET Problem Specification

Problem Type	Objective Criterion	
<input type="radio"/> Network Flow <input type="radio"/> Transportation Problem <input type="radio"/> Assignment Problem <input type="radio"/> Shortest Path Problem <input checked="" type="radio"/> Maximal Flow Problem <input type="radio"/> Minimal Spanning Tree <input type="radio"/> Traveling Salesman Problem	<input type="radio"/> Minimization <input checked="" type="radio"/> Maximization	
Data Entry Format		
<input checked="" type="radio"/> Spreadsheet Matrix Form <input type="radio"/> Graphic Model Form <input type="checkbox"/> Symmetric Arc Coefficients [i.e., both ways same cost]		
Problem Title	FLUJO MÁXIMO	
Number of Nodes	5	
OK	Cancel	Help

Network Modeling

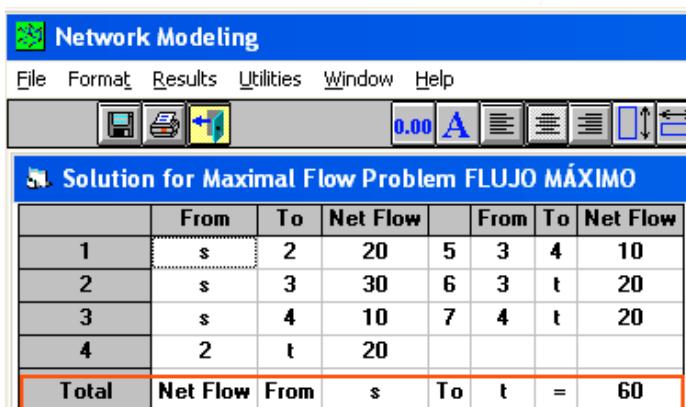
File Edit Format Solve and Analyze Results Utilities Window WinQSB Help

Maximal Flow Problem FLUJO MÁXIMO

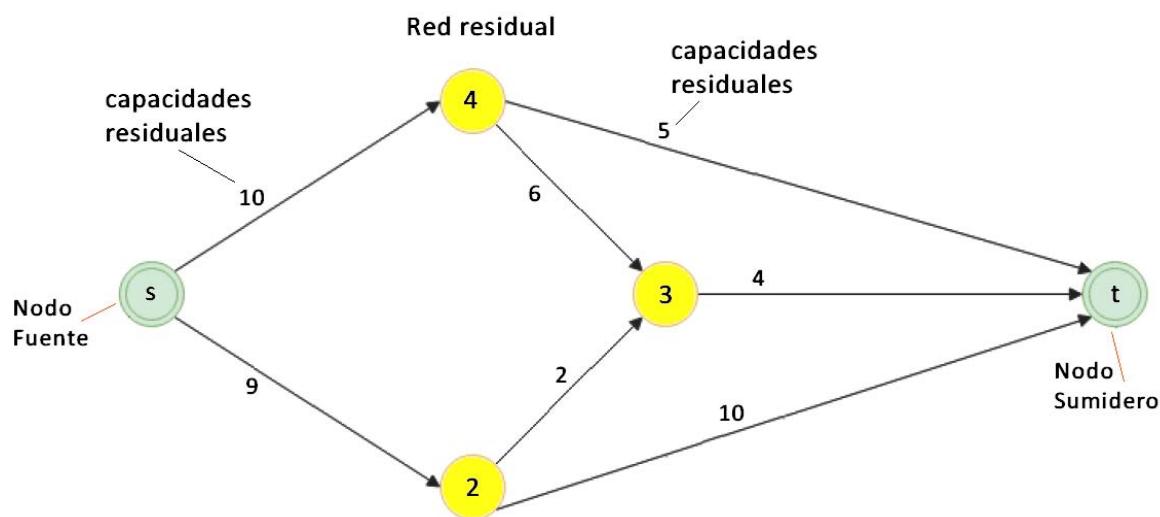
From \ To	s	2	3	4	t
s		20	30	10	
2			40		20
3				10	20
4			5		20
t					

Select Start and End Nodes

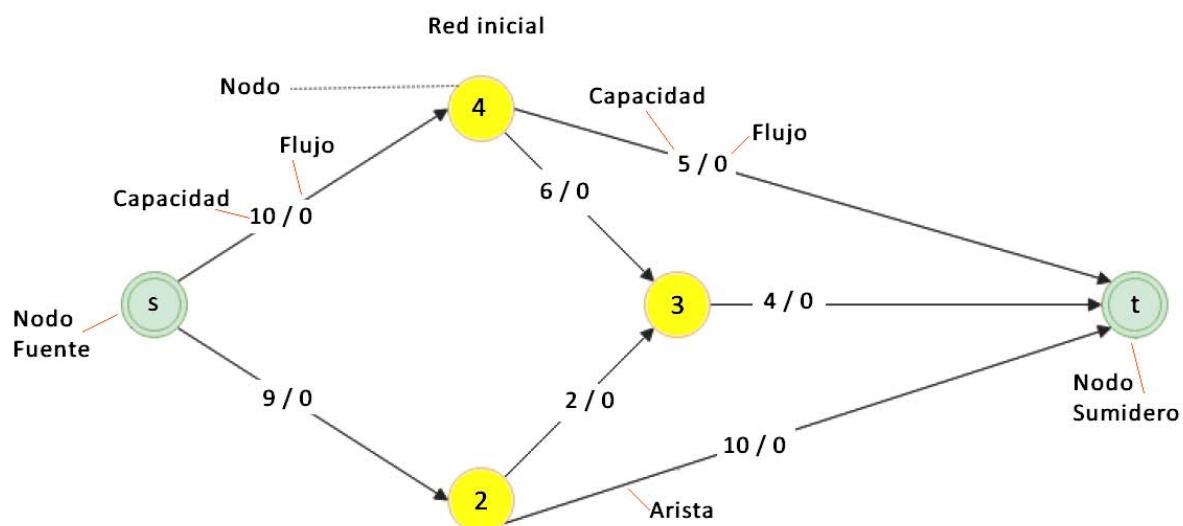
Click to select a start node	Click to select an end node
s 2 3 4 t	s 2 3 4 t
Solve	Solve and Display Steps
Cancel	Help



RESOLUCIÓN ALGORITMO DE FORD - FULKERSON



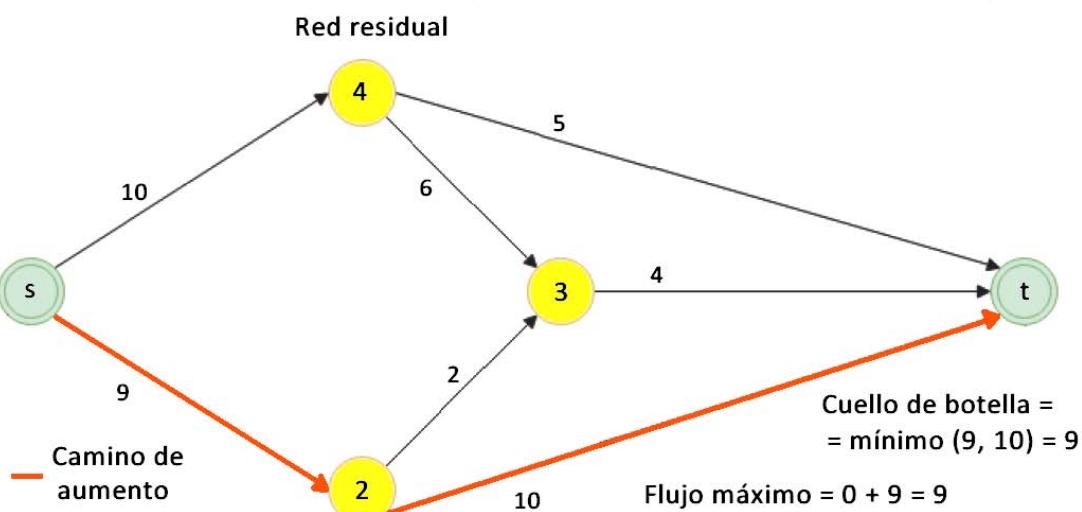
La red esta formada por 5 nodos, 7 aristas y una función capacidad. Se comienza poniendo todos los flujos a 0.



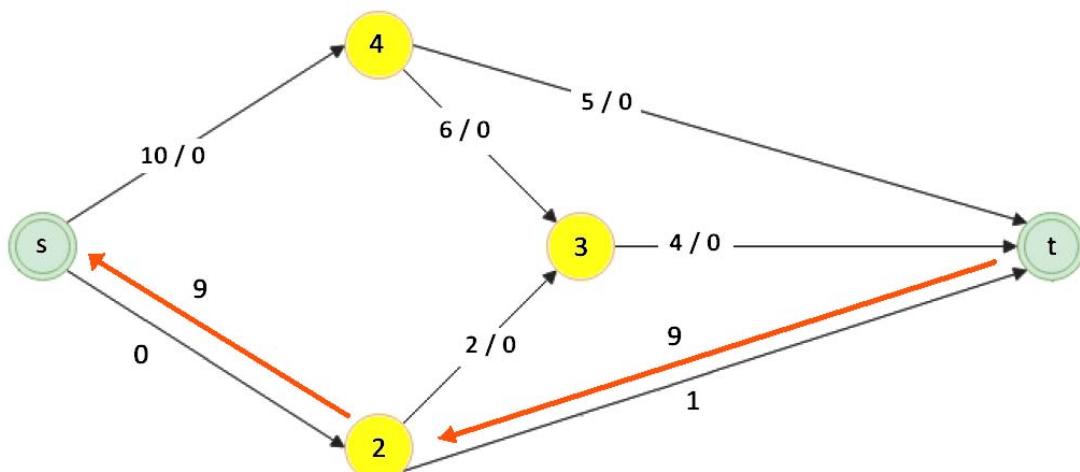
Camino de aumento: Camino sin nodos que une s con t

Cuello de botella: Mínimo de las capacidades residuales de un camino de aumento.

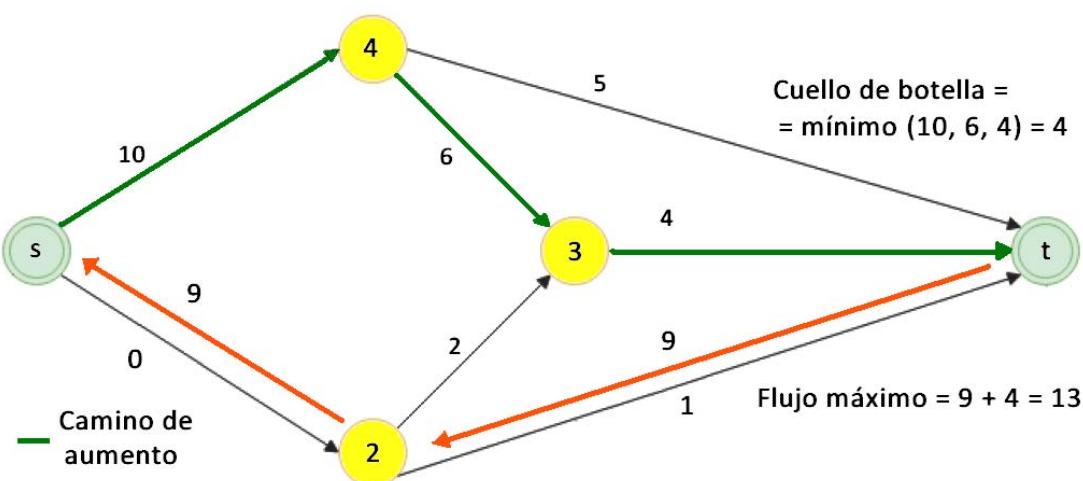
Se busca un camino de aumento: $s - 2 - t$



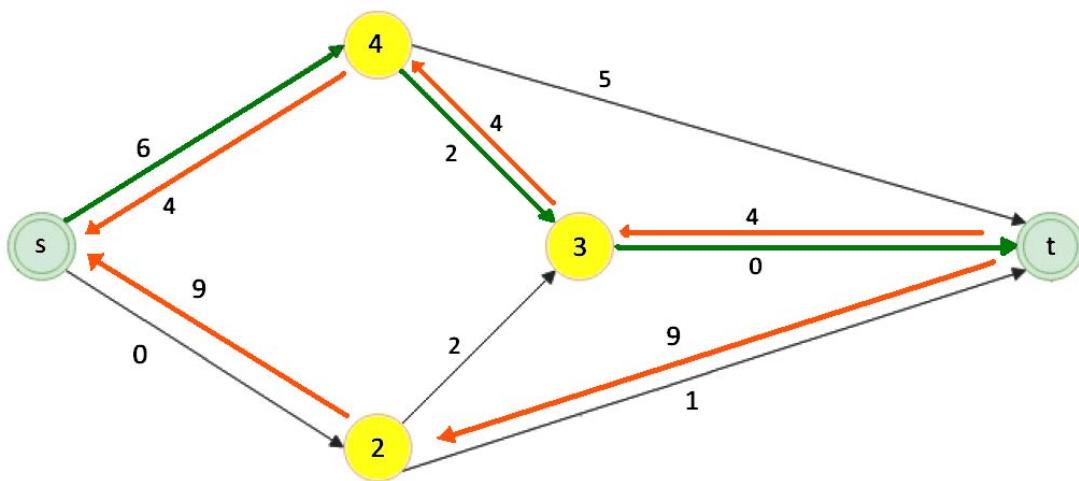
Se actualiza la red residual



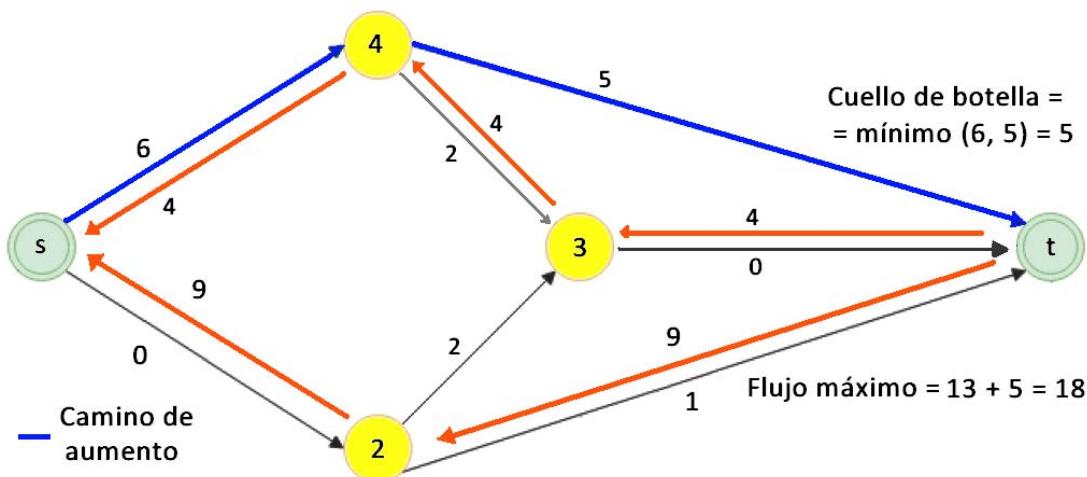
Se busca otro camino de aumento: $s - 4 - 3 - t$



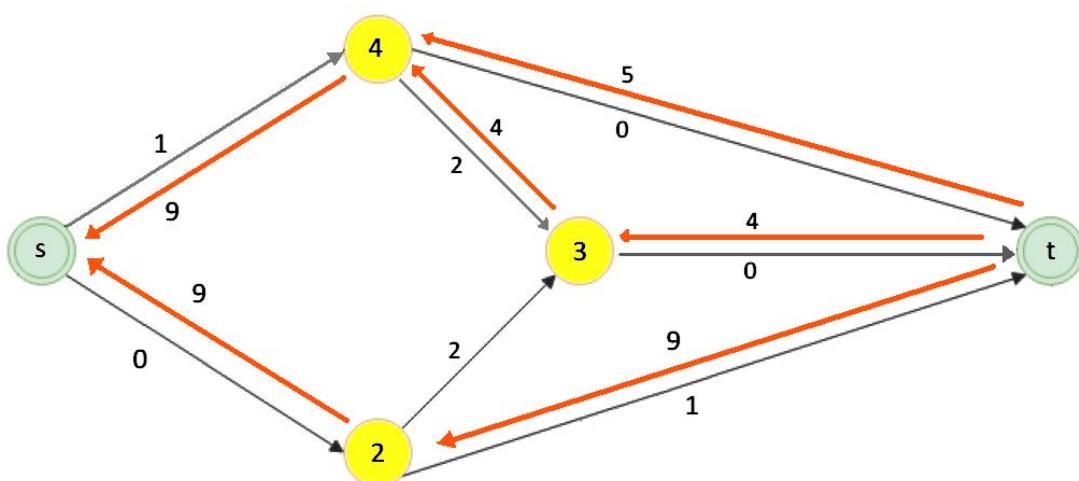
Se actualiza la red residual.



Se busca otro camino de aumento: $s - 4 - t$



Se actualiza la red residual.



El algoritmo de Ford - Fulkerson finaliza porque ya no se pueden encontrar más caminos en aumento. El flujo máximo es $\sum f(\text{arista}) = 9 + 9 = 18$



Network Modeling / Maximal Flow Problem

NET Problem Specification

Problem Type	Objective Criterion	
<input checked="" type="radio"/> Network Flow <input type="radio"/> Transportation Problem <input type="radio"/> Assignment Problem <input type="radio"/> Shortest Path Problem <input checked="" type="radio"/> Maximal Flow Problem <input type="radio"/> Minimal Spanning Tree <input type="radio"/> Traveling Salesman Problem	<input type="radio"/> Minimization <input checked="" type="radio"/> Maximization	
Data Entry Format		
<input checked="" type="radio"/> Spreadsheet Matrix Form <input type="radio"/> Graphic Model Form <input type="checkbox"/> Symmetric Arc Coefficients <small>(i.e., both ways same cost)</small>		
Problem Title	FLUJO MÁXIMO	
Number of Nodes	5	
OK	Cancel	Help

Network Modeling

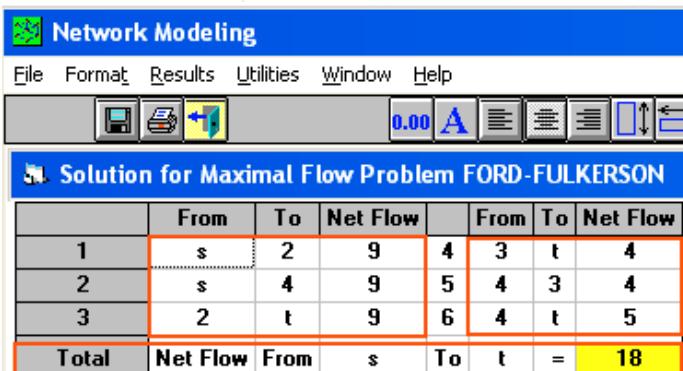
File Edit Format Solve and Analyze Results Utilities Window WinQSB Help

Maximal Flow Problem FORD-FULKERSON

From \ To	s	2	3	4	t
s		9		10	
2			2		10
3					4
4			6		5
t					

Select Start and End Nodes

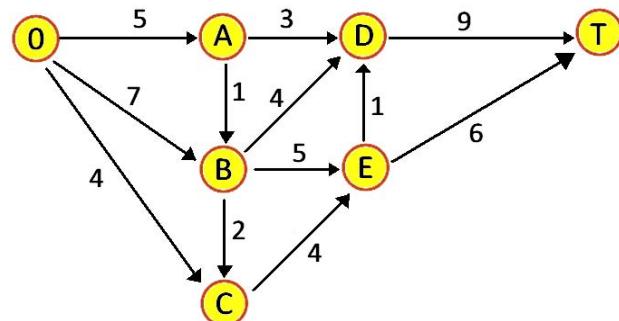
Click to select a start node	Click to select an end node
s 2 3 4 t	s 2 3 4 t
Solve	Solve and Display Steps
Cancel	Help



El Flujo máximo de s hasta t es 18.



Network Modeling / Maximal Flow Problem



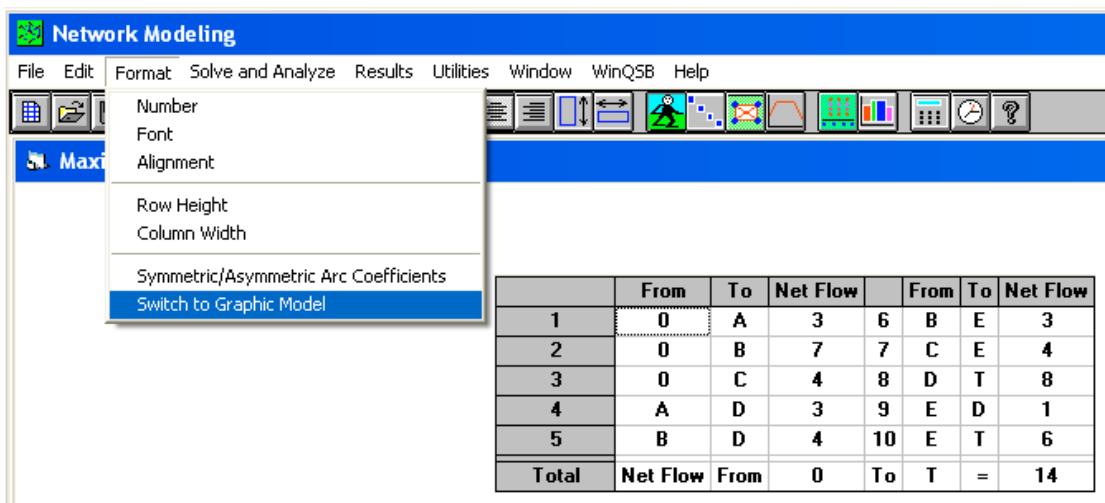
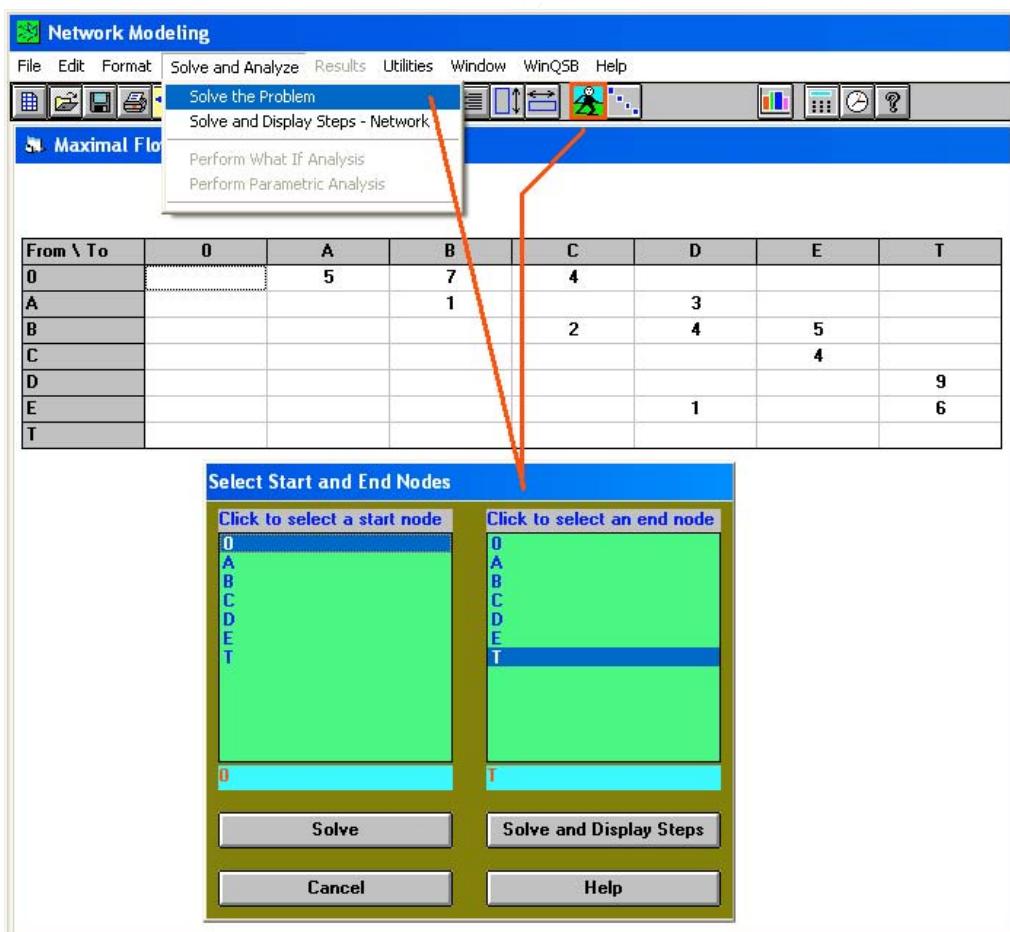
FORD - FULKERSON: Obtener el máximo flujo que se puede llevar del nodo 0 al nodo T

(IMPRESO EN PAPEL RECICLADO)

UNICAMENTE PARA USO ESCOLAR

NET Problem Specification

Problem Type	Objective Criterion	
<input type="radio"/> Network Flow <input type="radio"/> Transportation Problem <input type="radio"/> Assignment Problem <input type="radio"/> Shortest Path Problem <input checked="" type="radio"/> Maximal Flow Problem <input type="radio"/> Minimal Spanning Tree <input type="radio"/> Traveling Salesman Problem	<input type="radio"/> Minimization <input checked="" type="radio"/> Maximization	
Data Entry Format		
<input checked="" type="radio"/> Spreadsheet Matrix Form <input type="radio"/> Graphic Model Form <input type="checkbox"/> Symmetric Arc Coefficients (i.e., both ways same cost)		
Problem Title	FLUJO MÁXIMO	
Number of Nodes	7	
OK	Cancel	Help



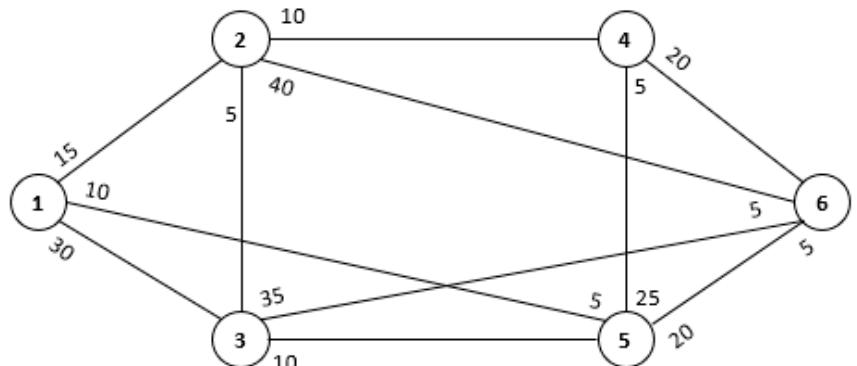
El Flujo máximo de 0 hasta T es 14.

FORD - FULKERSON: Una aerolínea española, con el objetivo de maximizar beneficios, ha contratado a un estudiante de Gestión Aeronáutica para realizar un estudio sobre el máximo flujo de vuelos que puede hacer en su ruta Madrid - Nueva York.

Según se refleja en el grafo, la aerolínea tiene vuelos en los siguientes destinos: Madrid - Barcelona - Las Palmas de Gran Canarias - Londres Heathrow - Miami - Nueva York.

Cada nodo representa el código de aeropuertos donde hace escala:

- 1 - Madrid (MAD)
- 2 - Barcelona (BCN)
- 3 - Las Palmas de Gran Canaria (LPA)
- 4 - Londres-Heathrow (LHR)
- 5 - Miami (MIA)
- 6 - Nueva York (JFK)



Calcula el flujo máximo por el Método de Ford-Fulkerson.

Solución:

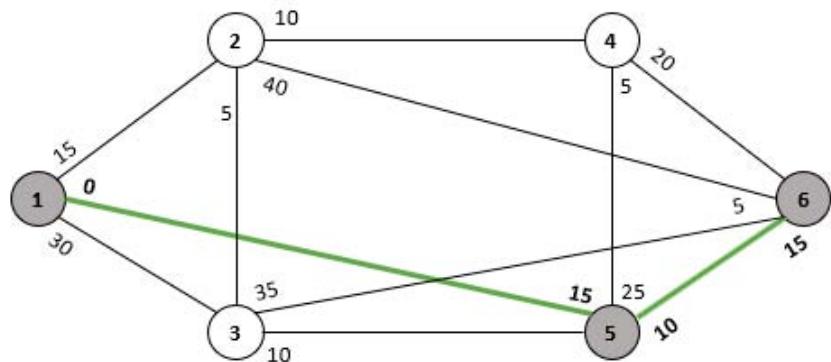
- Ruta: Madrid - Miami - Nueva York (1 - 5 - 6)

$$b = \min(10, 20) = 10$$

Flujo Máximo = $0 + 10 = 10$

$$C_{15,51} = (10 - 10, 5 + 10) = (0, 15)$$

$$C_{56,65} = (20 - 10, 5 + 10) = (10, 15)$$



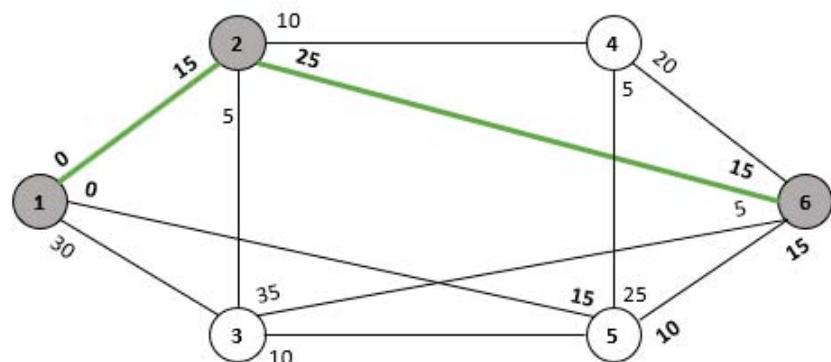
- Ruta: Madrid - Barcelona - Nueva York (1 - 2 - 6)

$$b = \min(15, 40) = 15$$

Flujo Máximo = $10 + 15 = 25$

$$C_{12,21} = (15 - 15, 0 + 15) = (0, 15)$$

$$C_{26,62} = (40 - 15, 0 + 15) = (25, 15)$$



■ Ruta: Madrid - Las Palmas - Miami - Londres - Nueva York (1 - 3 - 5 - 4 - 6)

$$b = \min(30, 10, 25, 20) = 10$$

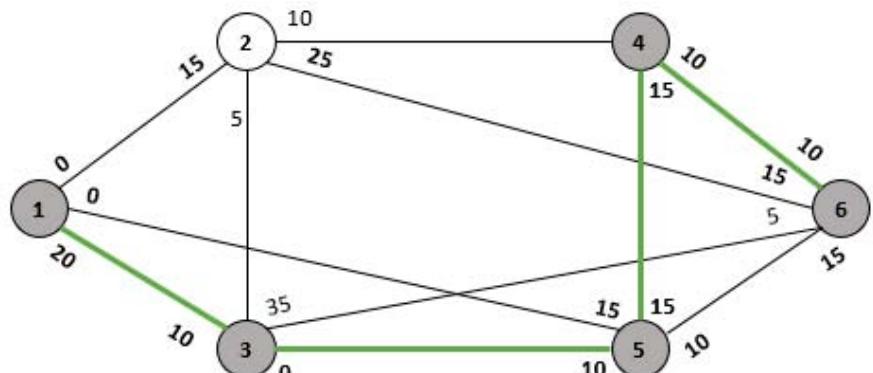
Flujo Máximo = $25 + 10 = 35$

$$C_{13,31} = (30 - 10, 0 + 10) = (20, 10)$$

$$C_{35,53} = (10 - 10, 0 + 10) = (0, 10)$$

$$C_{54,45} = (25 - 10, 5 + 10) = (15, 15)$$

$$C_{46,64} = (20 - 10, 0 + 10) = (10, 10)$$



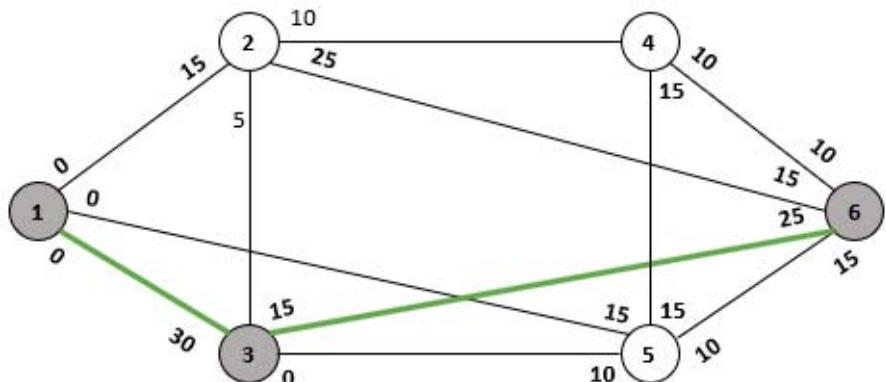
■ Ruta: Madrid - Las Palmas - Nueva York (1 - 3 - 6)

$$b = \min(20, 35) = 20$$

Flujo Máximo = $35 + 20 = 55$

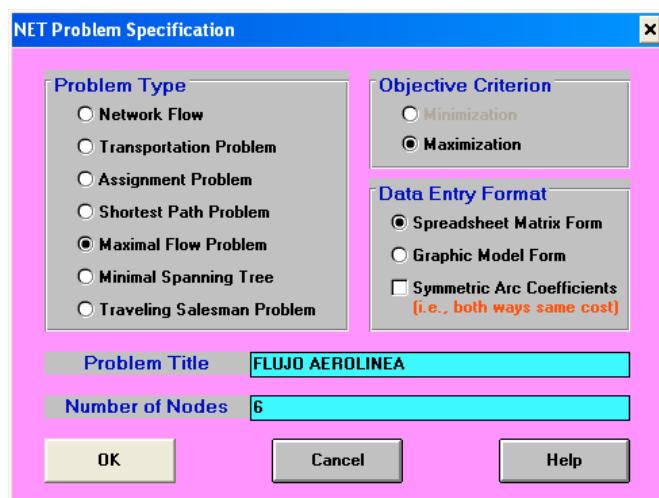
$$C_{13,31} = (20 - 20, 10 + 20) = (0, 30)$$

$$C_{36,63} = (35 - 20, 5 + 20) = (15, 25)$$



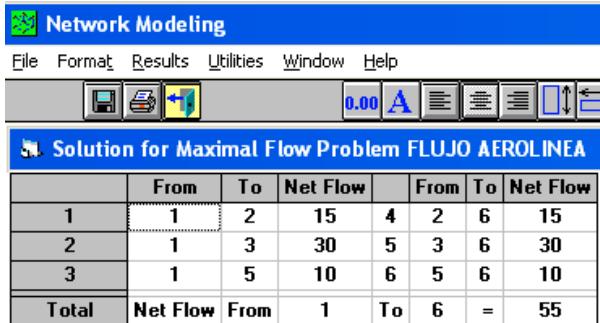
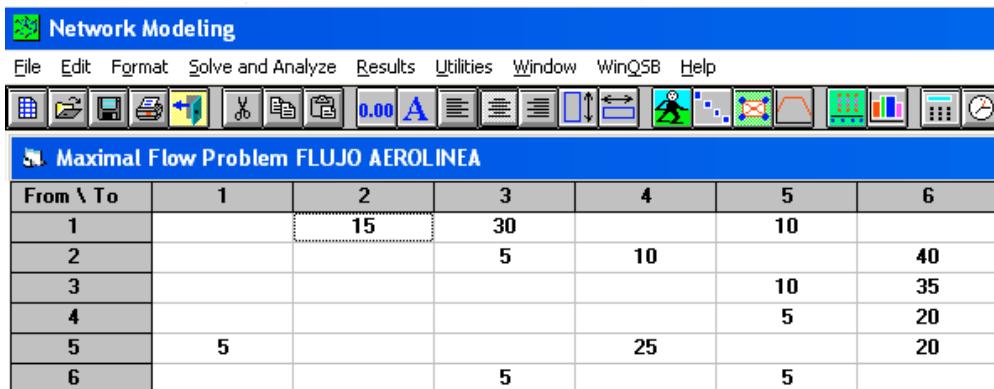
El algoritmo ha finalizado porque desde Madrid todos los aviones salen con capacidad 0.

El Flujo Máximo total de vuelos es 55.



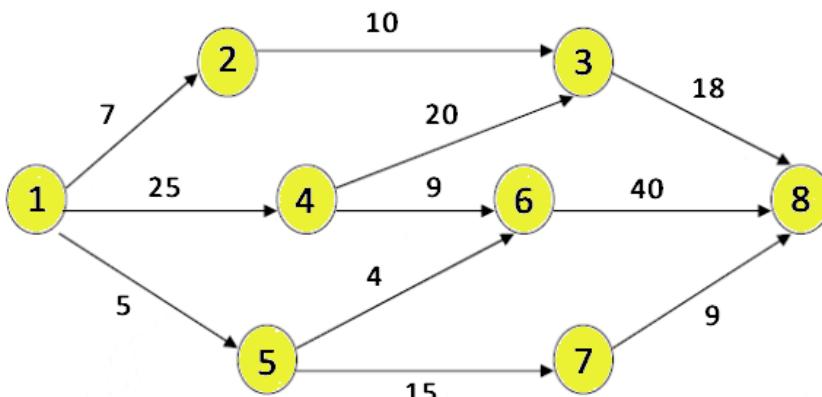
Resultado con WinQSB:

Network Modeling / Maximal Flow Problem



FLUJO MÁXIMO EN REDES: PROBLEMA DE PROGRAMACIÓN LINEAL

El Problema del Flujo Máximo es similar al Problema de Ruta más Corta, ahora se busca determinar el flujo máximo entre un nodo fuente y un nodo destino, que están enlazados a través de una red, con arcos de capacidad finita.



Los números asignados a cada uno de los arcos representan los flujos máximos o capacidades correspondientes a cada arco.

$x_{ij} \equiv$ Unidades que fluyen desde el nodo i al nodo j.

Función Objetivo: Maximizar las unidades que salen del nodo de origen (1) a los nodos que esta conectada (2, 4 y 5) o alternativamente maximizar las unidades que llegan al nodo de destino (8) desde los nodos que conectan a él (3, 6 y 7).

Alternativas: Maximizar $z = x_{12} + x_{14} + x_{15}$ o bien Maximizar $z = x_{38} + x_{68} + x_{78}$

Restricciones:

- Flujo Máximo: La cantidad de unidades que sale de cada nodo de origen a un nodo de destino no puede superar la capacidad del arco. Por ejemplo, del nodo 1 al nodo 2 sólo se pueden enviar 7 unidades.

$$\begin{cases} x_{12} \leq 7 & x_{14} \leq 25 & x_{15} \leq 5 & x_{23} \leq 10 & x_{38} \leq 18 \\ x_{43} \leq 20 & x_{46} \leq 9 & x_{56} \leq 4 & x_{57} \leq 15 & x_{68} \leq 40 & x_{78} \leq 9 \end{cases}$$

- Balance de Flujo en los Nodos: Las unidades que entran a un nodo debe de ser igual a las unidades que salen. Por ejemplo, el número de unidades que se envían del nodo 1 al nodo 4 debe ser igual a las que salen del nodo 4 (se envían al nodo 3 y al nodo 6).

$$\begin{array}{l} \left\{ \begin{array}{l} x_{12} = x_{23} \\ x_{14} = x_{43} + x_{46} \\ x_{15} = x_{56} + x_{57} \\ x_{23} + x_{43} = x_{38} \\ x_{46} + x_{56} = x_{68} \\ x_{57} = x_{78} \end{array} \right. \rightarrow \left\{ \begin{array}{l} x_{12} - x_{23} = 0 \\ x_{14} - x_{43} - x_{46} = 0 \\ x_{15} - x_{56} - x_{57} = 0 \\ x_{23} + x_{43} - x_{38} = 0 \\ x_{46} + x_{56} - x_{68} = 0 \\ x_{57} - x_{78} = 0 \end{array} \right. \end{array}$$

- No Negatividad e Integralidad: Las variables de decisión tienen la condición de no negatividad ($x_{ij} \geq 0$), adicionalmente se exige que éstas adopten valores enteros. Si se omite esta condición podría dar un problema de Programación Lineal.

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día



FLUJO MÁXIMO: PROGRAMACIÓN LINEAL

LP-ILP Problem Specification

Problem Title: FLUJO MÁXIMO

Number of Variables: 11 Number of Constraints: 6

Objective Criterion

Maximization Minimization

Default Variable Type

Nonnegative continuous Nonnegative integer Binary {0,1} Unsigned/unrestricted

Data Entry Format

Spreadsheet Matrix Form Normal Model Form

OK Cancel Help

Linear and Integer Programming

File Edit Format Solve and Analyze Results Utilities Window WinQSB Help

FLUJO MÁXIMO

Variable -->	X12	X14	X15	X23	X43	X46	X56	X57	X38	X68	X78	Direction	R. H. S.
Maximize	1	1	1										
C1	1			-1								=	0
C2		1			-1	-1						=	0
C3			1				-1	-1				=	0
C4				1	1				-1			=	0
C5						1	1			-1		=	0
C6								1			-1	=	0
LowerBound	0	0	0	0	0	0	0	0	0	0	0		
UpperBound	7	25	5	10	20	9	4	15	18	40	9		
VariableType	continuo												

FLUJO MÁXIMO

Maximize	1X12+1X14+1X15
OBJ/Constraint/VariableType/Bound	
Maximize	1X12+1X14+1X15
C1	1X12-1X23=0
C2	1X14-1X43-1X46=0
C3	1X15-1X56-1X57=0
C4	1X23+1X43-1X38=0
C5	1X46+1X56-1X68=0
C6	1X57-1X78=0
Integer:	
Binary:	
Unrestricted:	
X12	>=0, <=7
X14	>=0, <=25
X15	>=0, <=5
X23	>=0, <=10
X43	>=0, <=20
X46	>=0, <=9
X56	>=0, <=4
X57	>=0, <=15
X38	>=0, <=18
X68	>=0, <=40
X78	>=0, <=9

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

Linear and Integer Programming

File Format Results Utilities Window Help

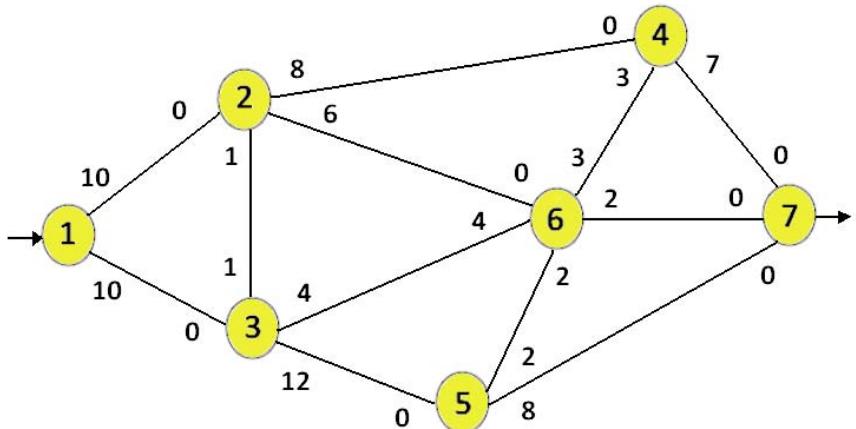
0.00 A

Solution Summary for FLUJO MÁXIMO

	Decision Variable	Solution Value	Unit Cost or Profit C(ij)	Total Contribution	Reduced Cost	Basis Status
1	X12	7	1	7	0	basic
2	X14	20	1	20	0	basic
3	X15	5	1	5	0	basic
4	X23	7	0	0	0	basic
5	X43	11	0	0	0	basic
6	X46	9	0	0	0	basic
7	X56	4	0	0	0	basic
8	X57	1	0	0	0	basic
9	X38	18	0	0	0	basic
10	X68	13	0	0	0	basic
11	X78	1	0	0	0	basic
Objective Function		(Max.) =	32	Note: Alternate	Solution Exists!	

El Flujo máximo que puede llegar al nodo de destino son 32 unidades. Los valores de las celdas en color amarillo representan la solución óptima, es decir, la cantidad de unidades que fluyen en cada combinación de un nodo origen - destino.

PROGRAMACIÓN LINEAL:
Encontrar el Flujo máximo de la red.



Función Objetivo: Maximizar las unidades que salen del nodo de origen (1) a los nodos que esta conectada (2 y 3) o alternativamente maximizar las unidades que llegan al nodo de destino (7) desde los nodos que conectan a él (4, 5 y 6).

Alternativas: Maximizar $z = x_{12} + x_{13}$ o bien Maximizar $z = x_{47} + x_{57} + x_{67}$

Restricciones:

- Flujo Máximo: La cantidad de unidades que sale de cada nodo de origen a un nodo de destino no puede superar la capacidad del arco.

$$\begin{cases} x_{12} \leq 10 & x_{13} \leq 10 & x_{23} \leq 1 & x_{32} \leq 1 & x_{26} \leq 6 & x_{36} \leq 4 & x_{63} \leq 4 & x_{24} \leq 8 \\ x_{64} \leq 3 & x_{46} \leq 3 & x_{35} \leq 12 & x_{65} \leq 2 & x_{56} \leq 2 & x_{57} \leq 8 & x_{47} \leq 7 & x_{67} \leq 2 \end{cases}$$

- Balance de Flujo en los Nodos: Las unidades que entran a un nodo debe de ser igual a las unidades que salen.

$$\text{Nodo 2: } x_{12} + x_{32} - x_{23} - x_{26} - x_{24} = 0$$

$$\text{Nodo 3: } x_{13} + x_{23} + x_{63} - x_{32} - x_{36} - x_{35} = 0$$

$$\text{Nodo 4: } x_{24} + x_{64} - x_{47} - x_{46} = 0$$

$$\text{Nodo 5: } x_{35} + x_{65} - x_{56} - x_{57} = 0$$

$$\text{Nodo 6: } x_{26} + x_{46} + x_{36} + x_{56} - x_{65} - x_{63} - x_{64} - x_{67} = 0$$

$$\text{Nodo 7: } x_{47} + x_{67} + x_{57} - x_{12} - x_{13} = 0$$

- No Negatividad e Integralidad: Las variables de decisión tienen la condición de no negatividad ($x_{ij} \geq 0$), adicionalmente se exige que éstas adopten valores enteros. Si se omite esta condición podría dar un problema de Programación Lineal.

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día



FLUJO MÁXIMO: PROGRAMACIÓN LINEAL

LP-ILP Problem Specification

Problem Title:	RED2
Number of Variables:	16
Number of Constraints:	6
Objective Criterion	<input checked="" type="radio"/> Maximization <input type="radio"/> Minimization
Default Variable Type	<input type="radio"/> Nonnegative continuous <input checked="" type="radio"/> Nonnegative integer <input type="radio"/> Binary (0,1) <input type="radio"/> Unsigned/unrestricted
Data Entry Format	<input checked="" type="radio"/> Spreadsheet Matrix Form <input type="radio"/> Normal Model Form
<input type="button" value="OK"/> <input type="button" value="Cancel"/> <input type="button" value="Help"/>	

Linear and Integer Programming

File Edit Format Solve and Analyze Results Utilities Window WinQSB Help

RED2

Variable -->	X12	X13	X23	X32	X26	X36	X63	X24	X64	X46	X35	X65	X56	X57	X47	X67	Direction	R. H. S.
Maximize	1	1															=	0
2	1		-1	1	-1				-1								=	0
3		1	1	-1			-1	1									=	0
4									1	1	-1						=	0
5											1	1	-1	-1			=	0
6								1	1	-1			-1	1			=	0
7	-1	-1								-1	1				1	1	=	0
LowerBound	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	=	0
UpperBound	10	10	1	1	6	4	4	8	3	3	12	2	2	8	7	2		
VariableType	integer																	

RED2

Maximize	1X12+1X13
OBJ/Constraint/VariableType/Bound	
Maximize	1X12+1X13
2	1X12-1X23+1X32-1X26-1X24=0
3	1X13+1X23-1X32-1X36+1X63-1X35=0
4	1X24+1X64-1X46-1X47=0
5	1X35+1X65-1X56-1X57=0
6	1X26+1X36-1X63-1X64+1X46-1X65+1X56-1X67=0
7	-1X12-1X13+1X57+1X47+1X67=0
Integer:	X12, X13, X23, X32, X26, X36, X63, X24, X64, X46, X35, X65, X56, X57, X47, X67
Binary:	
Unrestricted:	
X12	>=0, <=10
X13	>=0, <=10
X23	>=0, <=1
X32	>=0, <=1
X26	>=0, <=6
X36	>=0, <=4
X63	>=0, <=4
X24	>=0, <=8
X64	>=0, <=3
X46	>=0, <=3
X35	>=0, <=12
X65	>=0, <=2
X56	>=0, <=2
X57	>=0, <=8
X47	>=0, <=7
X67	>=0, <=2

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

Solution Summary for RED2						
	Decision Variable	Solution Value	Unit Cost or Profit C(i)	Total Contribution	Reduced Cost	Basis Status
1	X12	9	1	9	0	basic
2	X13	8	1	8	0	basic
3	X23	0	0	0	0	at bound
4	X32	0	0	0	0	at bound
5	X26	2	0	0	0	basic
6	X36	0	0	0	0	at bound
7	X63	0	0	0	0	at bound
8	X24	7	0	0	0	basic
9	X64	0	0	0	0	at bound
10	X46	0	0	0	0	at bound
11	X35	8	0	0	0	basic
12	X65	0	0	0	0	at bound
13	X56	0	0	0	0	at bound
14	X57	8	0	0	0	basic
15	X47	7	0	0	0	basic
16	X67	2	0	0	0	basic
		Objective Function	(Max.) =	17		

El Flujo máximo que puede llegar al nodo de destino son 17 unidades. Los valores de las celdas en color amarillo representan la solución óptima, es decir, la cantidad de unidades que fluyen en cada combinación de un nodo origen - destino.

ALGORITMO DE PRIM: ÁRBOL DE EXPANSIÓN MÍNIMA

Se utiliza para encontrar un árbol recubridor mínimo en un grafo conexo, no dirigido y con aristas etiquetadas.

El algoritmo encuentra un subconjunto de aristas que forman un árbol con todos los vértices, donde el peso total de las aristas en el árbol es el mínimo posible.

Si el grafo no es conexo, el algoritmo encuentra el árbol recubridor mínimo para una de las componentes conexas (que forman el grafo no conexo).

El algoritmo va incrementando continuamente el tamaño del árbol, comienza con un vértice inicial (elegido arbitrariamente) al que se van añadiendo sucesivamente vértices cuya distancia a los anteriores es mínima.

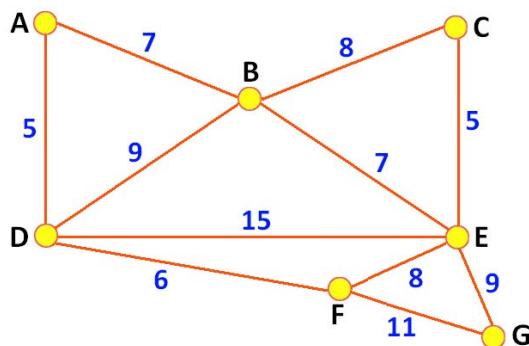
Esto conlleva que en cada paso, las aristas a considerar son aquellas que inciden en vértices que ya pertenecen al árbol.

El algoritmo termina cuando el árbol recubridor mínimo está completamente construido, es decir, cuando no quedan más vértices que agregar.

Informalmente el algoritmo se puede describir:

- Iniciar un árbol con un único vértice, elegido arbitrariamente del grafo.
- Aumentar el árbol por un lado. Un lado es la unión entre dos vértices: De las posibles uniones que pueden conectar el árbol a los vértices que no están aún en el árbol, encontrar el lado de menor distancia y unirlo al árbol.
- Repetir la secuencia (hasta que todos los vértices pertenezcan al árbol)

Calcular el peso que transporta el árbol de expansión mínima del grafo adjunto



Inicialmente el grafo ponderado no es un árbol (para serlo se requiere que no hubiera ciclos y en este caso sí hay).

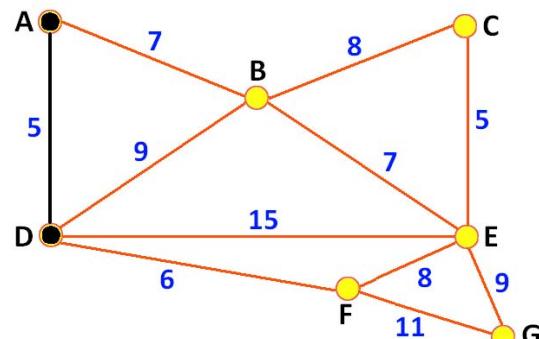
Comienza el proceso eligiendo arbitrariamente al vértice D como punto de partida.

Los números cerca de las aristas indican el peso.

Se elige arbitrariamente el vértice D como punto de partida.

El vértice más cercano a D es el vértice A, a una distancia de 5.

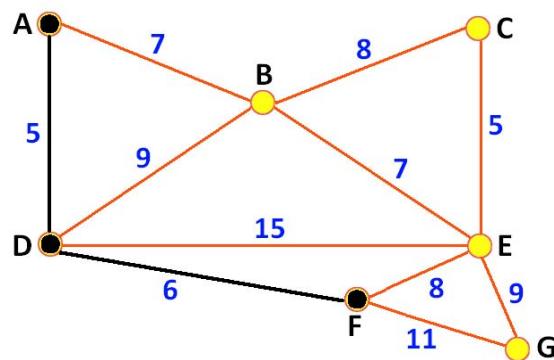
Se marca la arista DA.



El vértice más cercano al vértice D o al vértice A es el vértice F, a una distancia de 6.

Adviértase que el vértice B está a una distancia de 7 del vértice A.

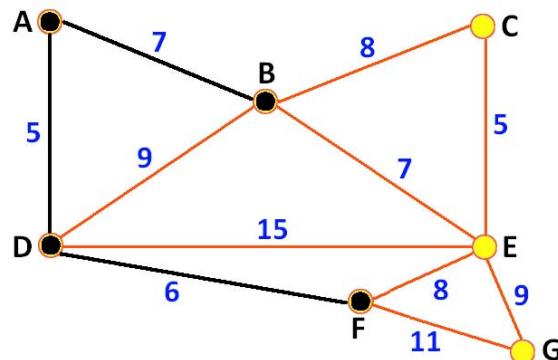
Se marca la arista DF.



El algoritmo continúa, el vértice más cercano a los vértices A y F es el vértice B, a una distancia 7 del vértice A.

Se marca la arista AB.

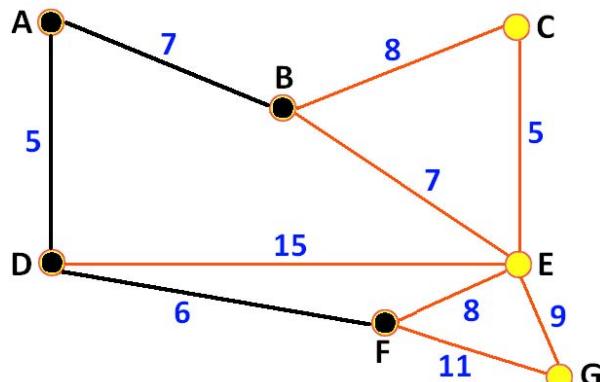
En esta situación, la arista DB se anula porque sus dos extremos ya están en el árbol y, en consecuencia, no podrá ser utilizada.

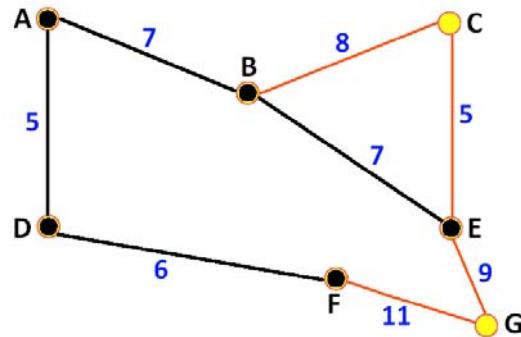


Los vértices más cercanos a B y F son los vértices C, E y G.

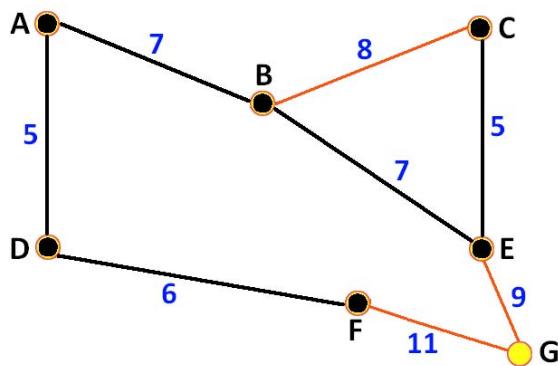
Se elige el vértice E que se encuentra a una distancia 7 de B.

Se marca la arista BE.





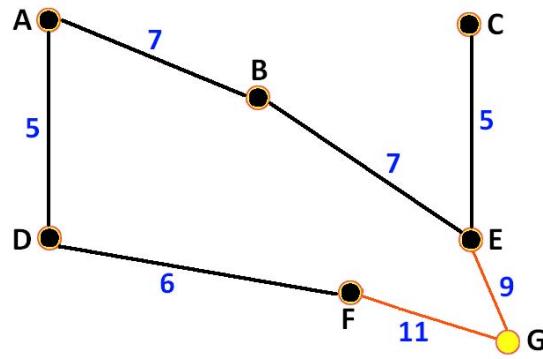
Desaparecen otras dos aristas, ED y EF porque ambos vértices que unen fueron agregados al árbol.



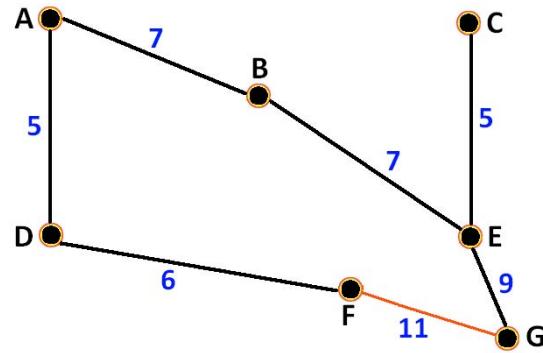
Los vértices más cercanos a E y F son C y G.

Se elige el vértice C que se encuentra a una distancia 5 de del vértice E.

Se marca la arista EC.



Desaparece la arista BC porque los vértices fueron agregados al árbol.



El único vértice pendiente G se encuentra más cercano al vértice E, a una distancia de 9.

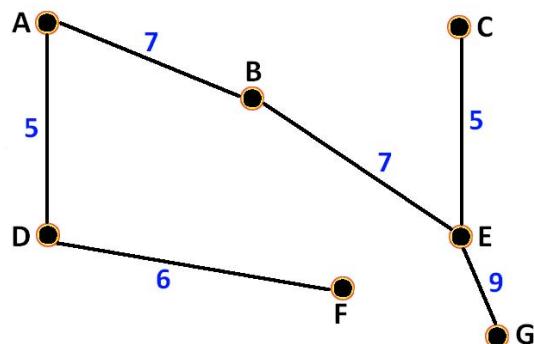
Se marca la arista EG.

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

Desaparece la arista FG porque los vértices fueron agregados al árbol.

El algoritmo termina al estar marcados todos los vértices.

El árbol de expansión mínima tiene un peso de 39.



PRIM: Network Modeling / Minimal Spanning Tree

NET Problem Specification

Problem Type	Objective Criterion	
<input type="radio"/> Network Flow <input type="radio"/> Transportation Problem <input type="radio"/> Assignment Problem <input type="radio"/> Shortest Path Problem <input type="radio"/> Maximal Flow Problem <input checked="" type="radio"/> Minimal Spanning Tree <input type="radio"/> Traveling Salesman Problem	<input checked="" type="radio"/> Minimization <input type="radio"/> Maximization	
Data Entry Format		
<input checked="" type="radio"/> Spreadsheet Matrix Form <input type="radio"/> Graphic Model Form <input checked="" type="checkbox"/> Symmetric Arc Coefficients <small>[i.e., both ways same cost]</small>		
Problem Title	PRIM-1	
Number of Nodes	7	
OK	Cancel	Help

Network Modeling

File Edit Format Solve and Analyze Results Utilities Window WinQSB Help

0.00 A

Minimal Spanning Tree Problem PRIM-1

From \ To	A	B	C	D	E	F	G
A		7		5			
B	7		8	9	7		
C		8			5		
D	5	9			15	6	
E		7	5	15		8	9
F				6	8		11
G					9	11	

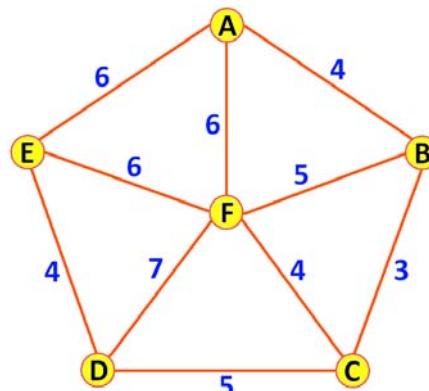
Network Modeling

File Format Results Utilities Window Help

0.00 A

Solution for Minimal Spanning Tree Problem PRIM-1

	From Node	Connect To	Distance/Cost		From Node	Connect To	Distance/Cost
1	A	B	7	4	B	E	7
2	E	C	5	5	D	F	6
3	A	D	5	6	E	G	9
	Total	Minimal	Connected	Distance	or Cost	=	39



- PRIM: Construir un árbol generador del grafo ponderado adjunto utilizando el algoritmo de Prim.

El algoritmo de Prim comienza con un vértice cualquiera, se elige arbitrariamente el vértice A y se construye el árbol T con A como único vértice.

En pasos sucesivos, se va añadiendo al árbol parcial T la arista de menor peso que une un vértice de T con un vértice que todavía no está en T.

Cuando una arista tenga sus dos extremos en el árbol no podrá ser utilizada.

El algoritmo termina cuando todos los vértices se encuentren seleccionados, es decir, cuando se han añadido $(n - 1)$ aristas.

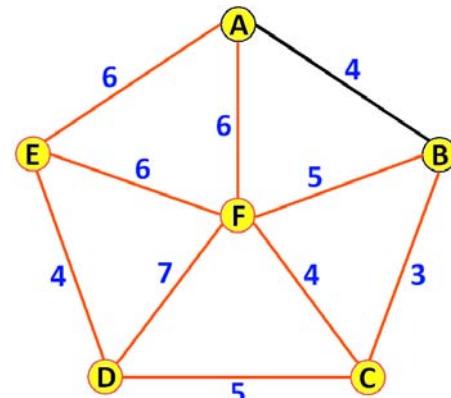
Finalmente, el árbol de expansión mínimo lleva un peso.

El grafo ponderado de partida no es un árbol, ya que para serlo requiere que no tuviera ciclos.

Los números cerca de las aristas indican los pesos.

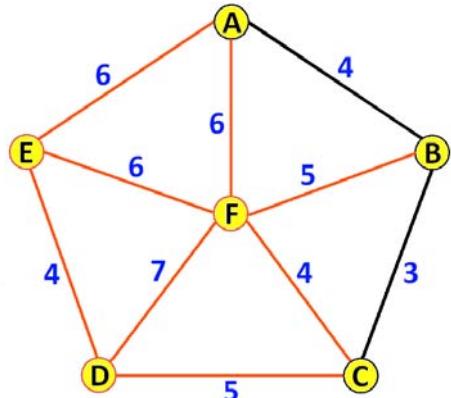
Se ha elegido arbitrariamente el vértice A como punto de partida. El segundo vértice más cercano al vértice A que se encuentra a menor distancia es el vértice B.

Se marca la arista AB.

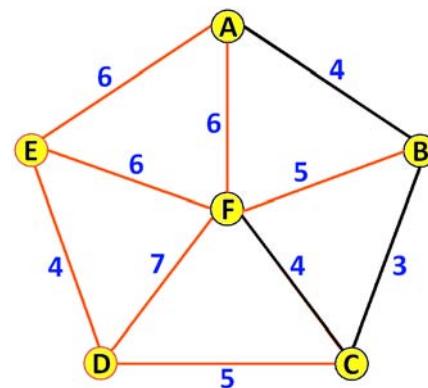


El próximo vértice a elegir que sea más cercano al vértice A o al vértice B es el vértice C, a una distancia 3.

Se marca la arista BC.



El algoritmo continúa. El vértice F, que está a una distancia 4 de C, es el siguiente marcado.

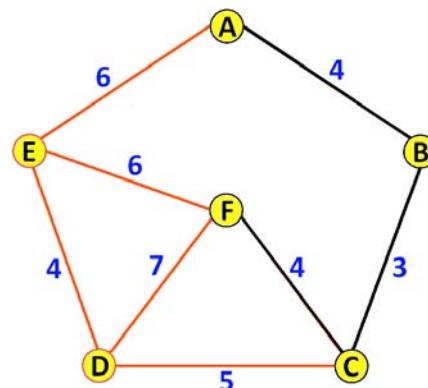


En este punto la arista BF desaparece porque ambos vértices fueron agregados al árbol T.

En esta línea, también desaparece la arista AF, al ser los vértices agregados al árbol.

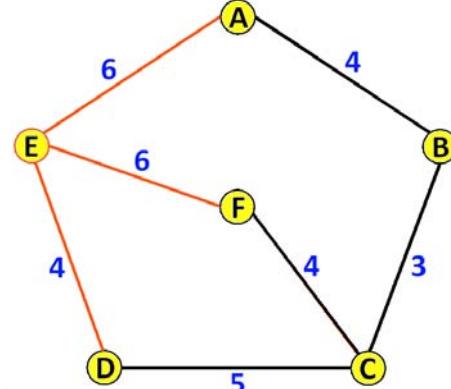
Aquí hay que elegir entre C (distancia 5 de D), o los vértices D o E a distancias respectivas del vértice F de 7 o 6.

Se marca el vértice CD.



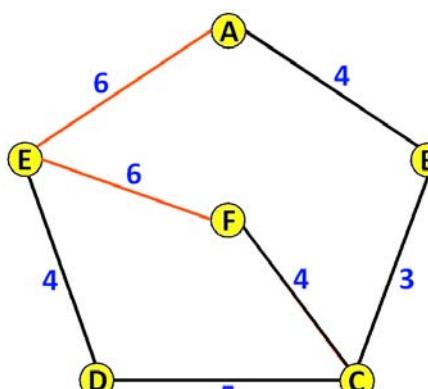
En este punto la arista DF desaparece porque ambos vértices fueron agregados al árbol T.

$$V(T) = \{A, B, C, F, D\}$$



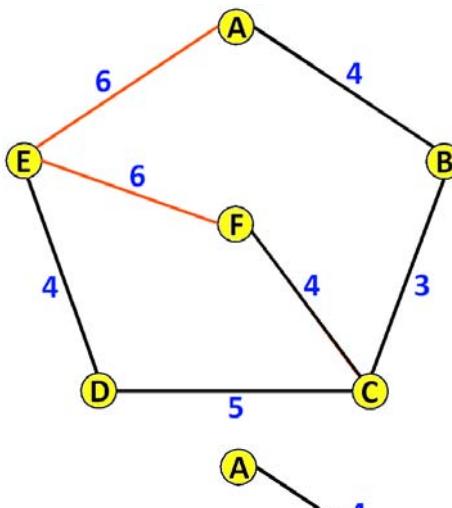
Se elige el vértice E que se encuentra a una distancia 4 de D.

Se marca el vértice DE.



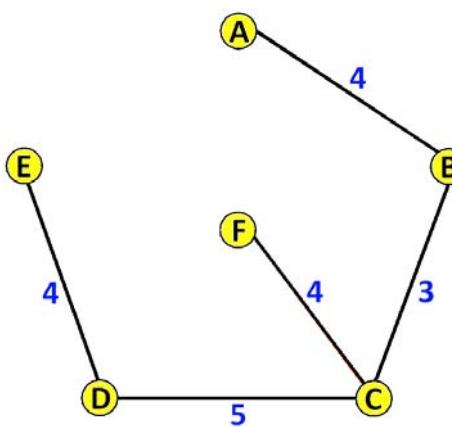
La arista EF desaparece porque ambos vértices fueron agregados al árbol T.

Por el mismo criterio desaparece la arista EA.



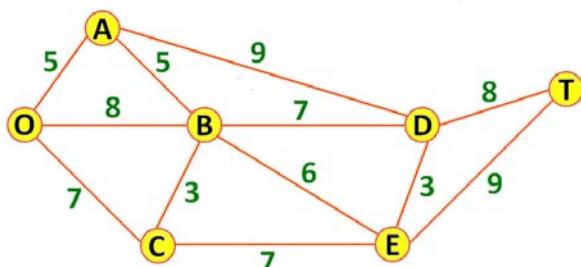
El árbol generador de expansión mínima se muestra en la figura adjunta. En este caso con un peso de 20.

$$V(T) = \{A, B, C, D, E, F\}$$



Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

PRIM: Encontrar el árbol de expansión mínima.

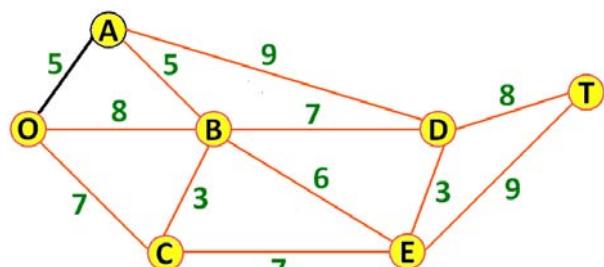


El algoritmo de Prim comienza con un vértice cualquiera, se elige arbitrariamente el vértice O y se construye el árbol T con O como único vértice.

En pasos sucesivos, se va añadiendo al árbol parcial T la arista de menor peso que une un vértice de T con un vértice que todavía no está en T. Cuando una arista tenga sus dos extremos en el árbol no podrá ser utilizada.

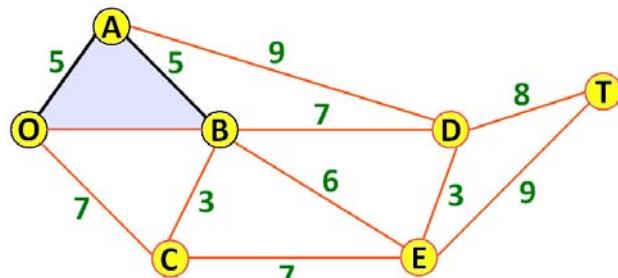
El segundo vértice más cercano a O es el vértice A (a una distancia de 5), el vértice C está a una distancia de 7.

Se marca la arista OA.



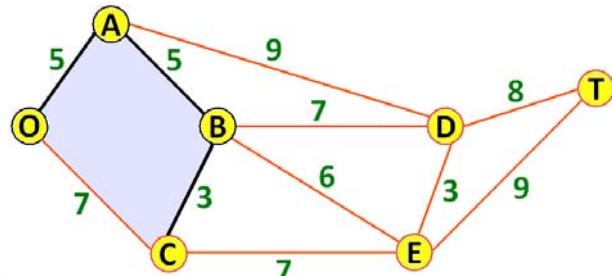
El siguiente vértice más cercano a elegir entre O o A es el vértice B, a una distancia 5.

Se marca la arista AB, mientras que la arista OB desaparece porque formaría el ciclo OAB.



El siguiente vértice más cercano a elegir entre O o B es el vértice C, a una distancia 3.

Se marca la arista BC, desaparece la arista OC porque formaría el ciclo OABC.

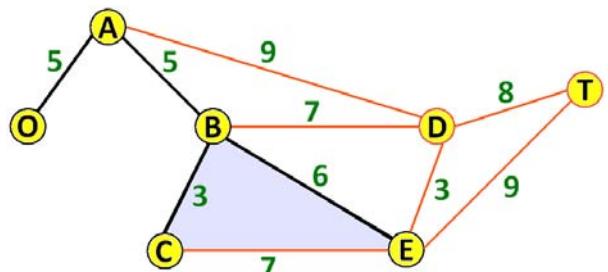


Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

El vértice más cercano a B o C es el vértice E (distancia 6) del vértice B.

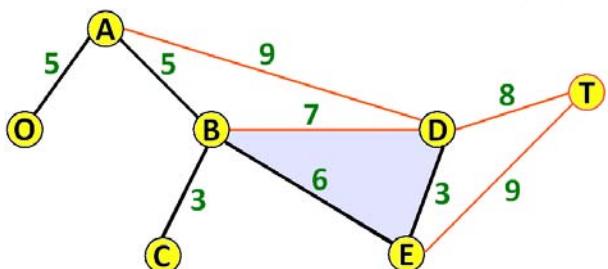
Se marca la arista BE.

Desaparece la arista CE porque formaría el ciclo BCE.

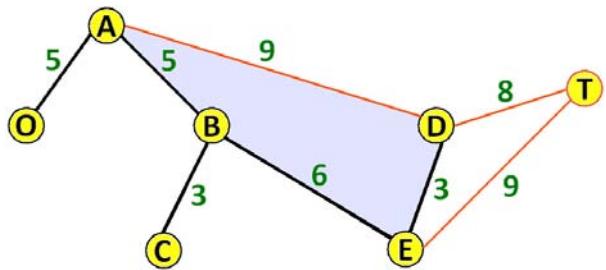


El siguiente vértice más cercano a elegir entre A y E es el vértice D, a una distancia 3 del vértice E.

Se marca la arista ED, desaparece la arista BD porque formaría el ciclo BED.

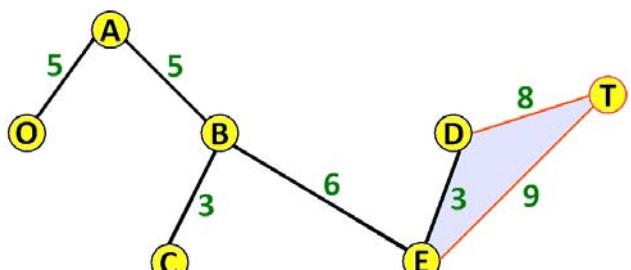


También desaparece la arista AD porque formaría el ciclo ABED.

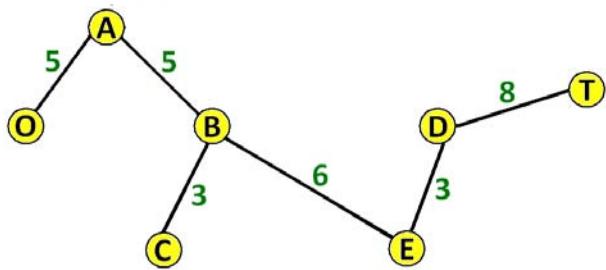


El siguiente vértice más cercano a elegir entre D y E es el vértice T, a una distancia 8 del vértice T.

Desaparece la arista ET porque formaría el ciclo DET.

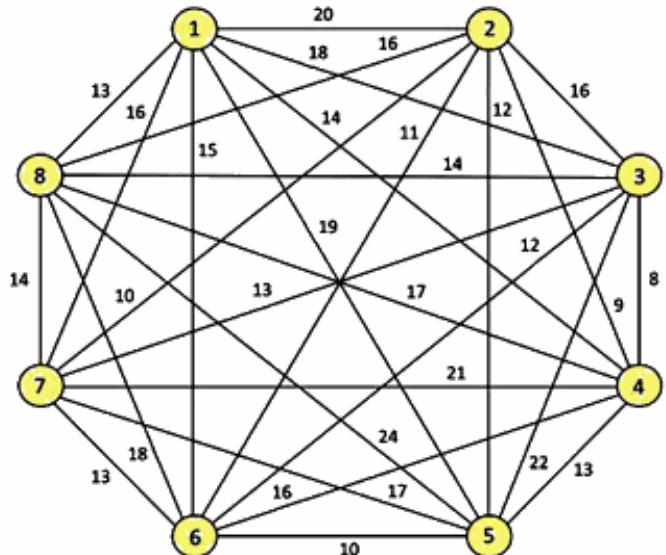


El algoritmo termina, todos los vértices están marcados, y se obtiene el árbol de expansión mínima con un peso de 30.



PRIM (ÁRBOL EXPANSIÓN MÍNIMA)

Una empresa aeronáutica decide instalar un sistema de distribución que permita enviar contenedores a ocho provincias españolas, considerando la posibilidad de envío no directo. Los costes en cientos de euros se reflejan en la tabla adjunta. Calcular el mínimo coste.

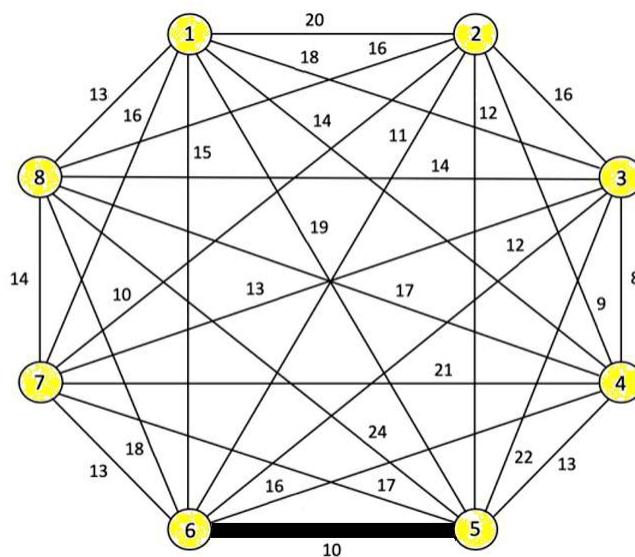


El algoritmo de Prim comienza eligiendo un vértice cualquiera.

Se elige arbitrariamente el vértice 5 y se construye el árbol con 5 como único vértice.

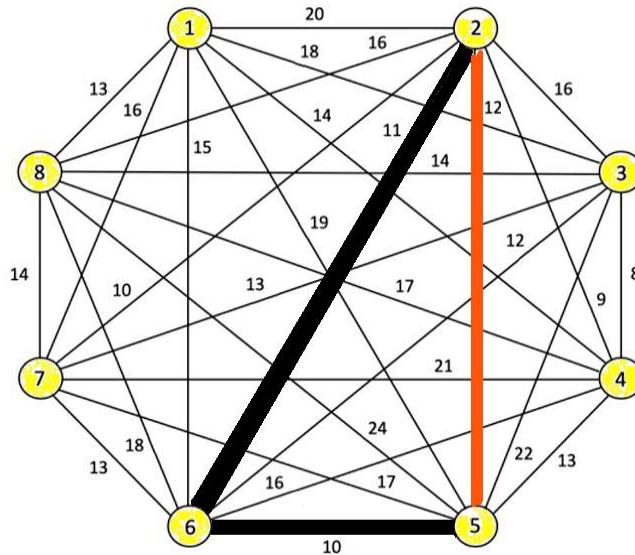
A continuación, se añade la arista de menor peso que une el vértice.

El segundo vértice más cercano a 5 es el vértice 6 (a una distancia de 10). Se marca la arista 5-6.

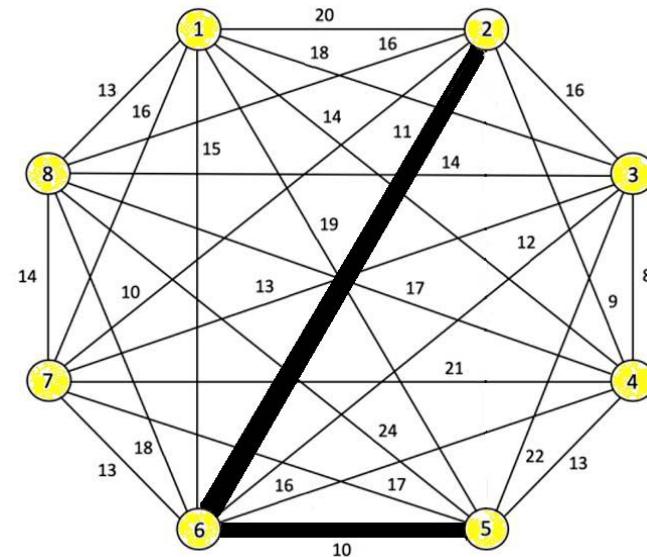


El siguiente vértice más cercano a elegir entre 5 y 6 es el vértice 2, a una distancia de 11 del vértice 6.

Se marca la arista 6-2, mientras que la arista 2-5 desaparece porque formaría un ciclo.

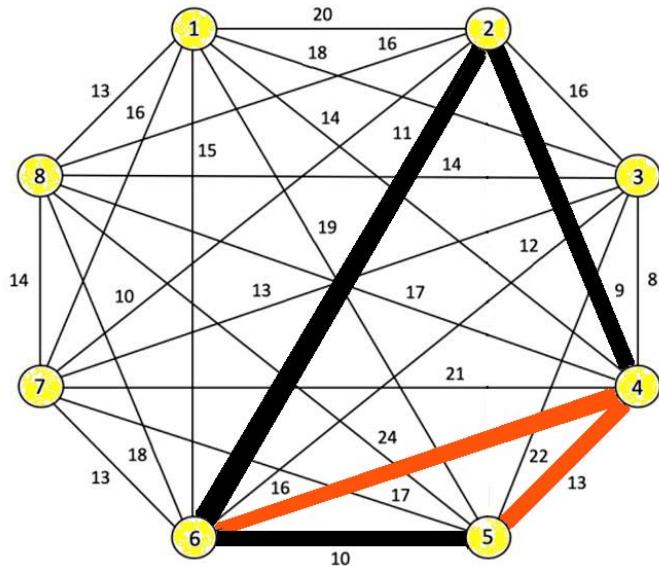
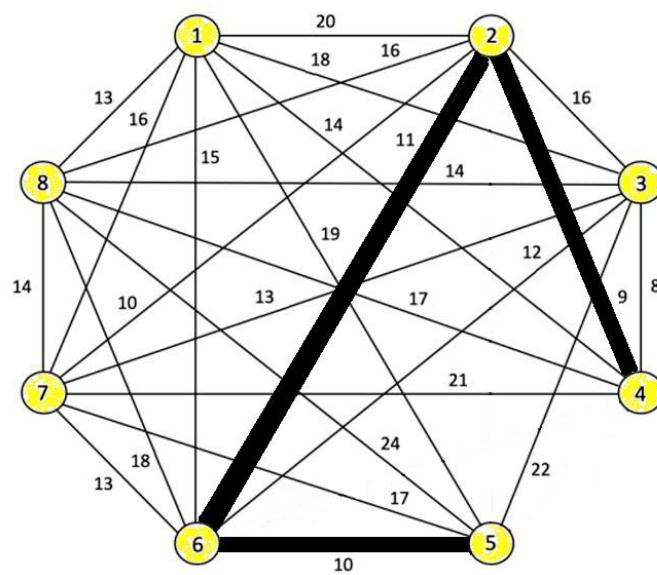


Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

Grafo resultante:

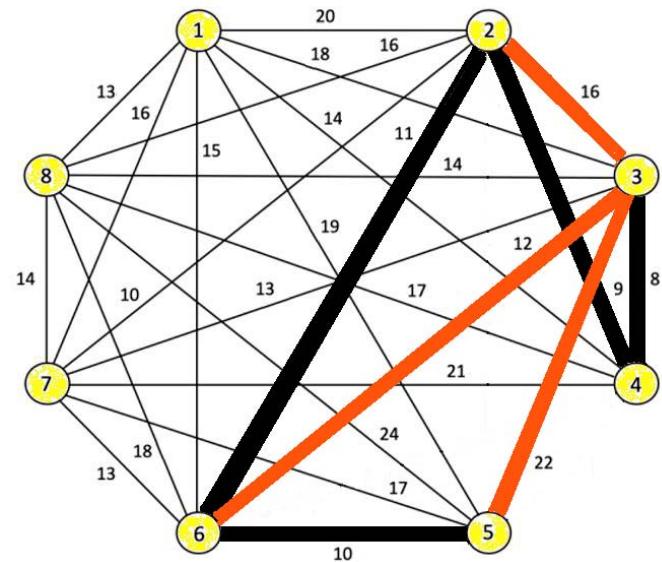
El siguiente vértice más cercano a elegir entre 6 o 2 es el vértice 4, a una distancia de 9 del vértice 2.

Se marca la arista 2-4 y desaparecen las aristas 4-6 y 4-5 porque formarían un ciclo

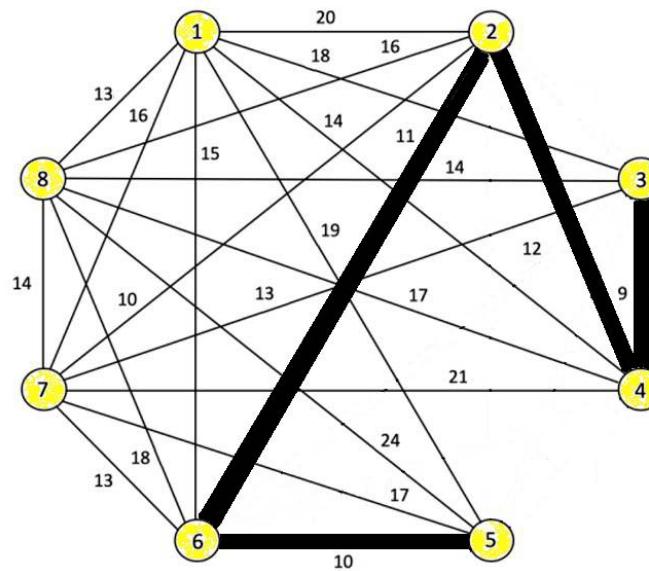
**Grafo resultante:**

El vértice más cercano a 2 o 4 es el vértice 3, a una distancia de 8 del vértice 4.

Se marca la arista 4-3 a una distancia de 8 y desaparece las aristas 3-2 , 3-6 y 3-5 para no formar ciclos.

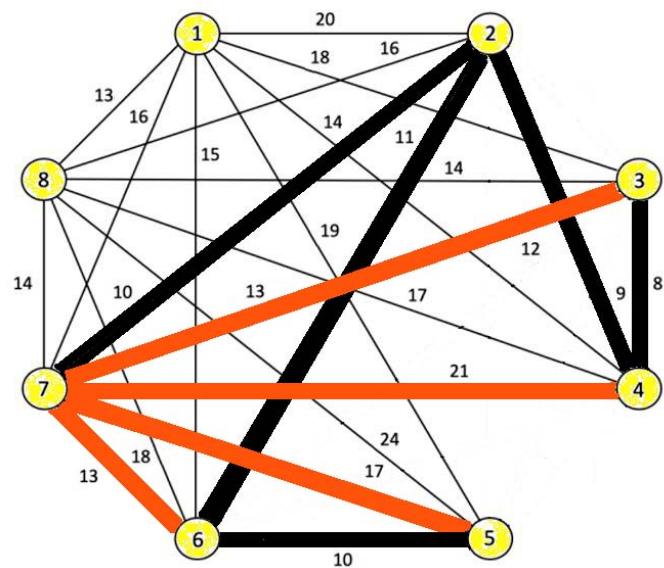


Grafo resultante:

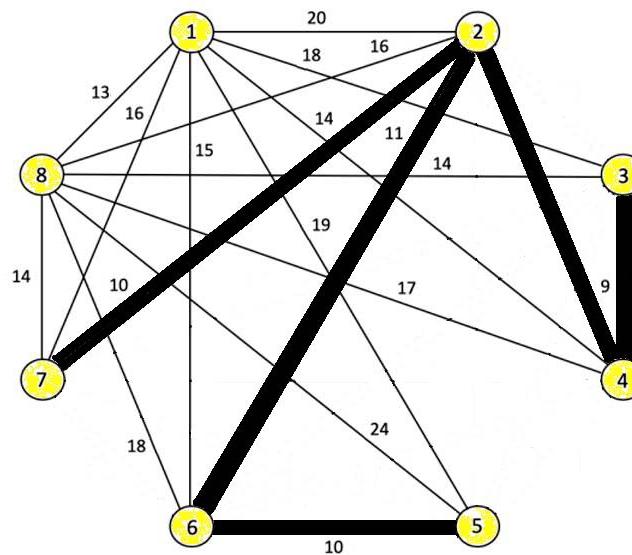


El siguiente vértice más cercano a elegir entre 2 o 3 es el vértice 7, a una distancia de 10 del vértice 2.

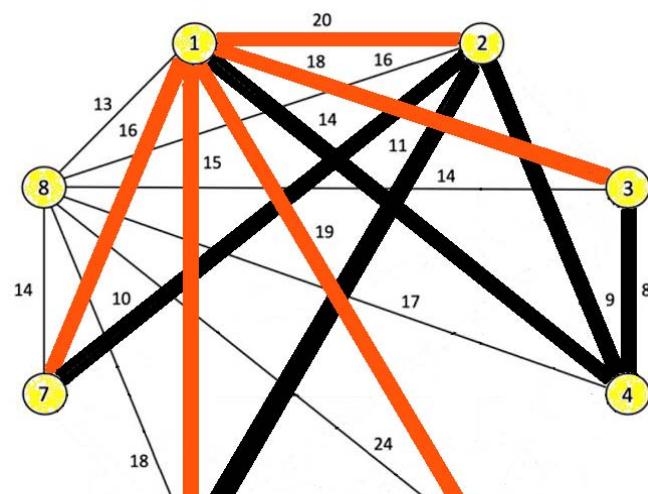
Se marca la arista 2-7 y desaparecen las aristas 7-3 , 7-6 , 7-5 y 7-4 para no formar ciclos.



Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

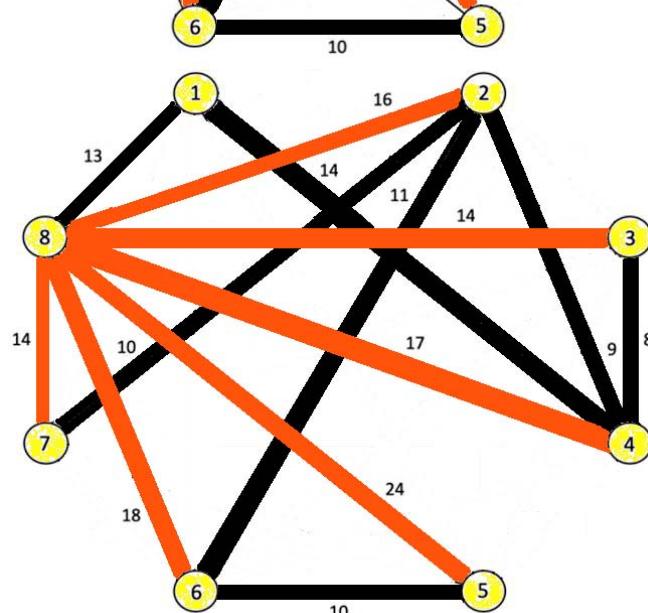


Grafo resultante:



El siguiente vértice más cercano es el 1, a una distancia de 14 del vértice 4.

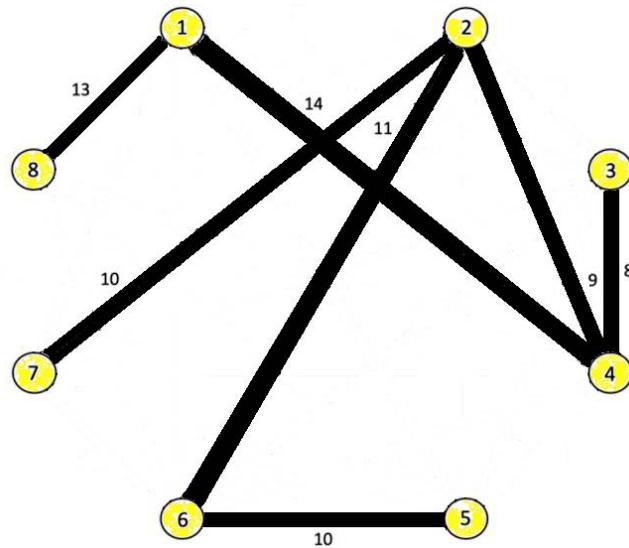
Se marca la arista 4-1 y desaparecen las aristas 1-7, 1-3, 1-2, 1-5 y 1-6 para no formar ciclos.



Finalmente, se elige el vértice 8, a una distancia de 13 del vértice 1.

Se marca la arista 1-8 y desaparecen las aristas 8-2, 8-5, 8-6, 8-3, 8-4 y 8-7 para no formar ciclos

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día



El algoritmo finaliza, todos los vértices están marcados.

Longitud o peso mínimo de la distribución:

$$8 + 9 + 10 + 10 + 11 + 13 + 14 = 75$$

Coste mínimo de la distribución:

$$75 \times 100 = 7.500 \text{ euros.}$$

ALGORITMO DE KRUSKAL: ÁRBOL DE EXPANSIÓN MÍNIMA

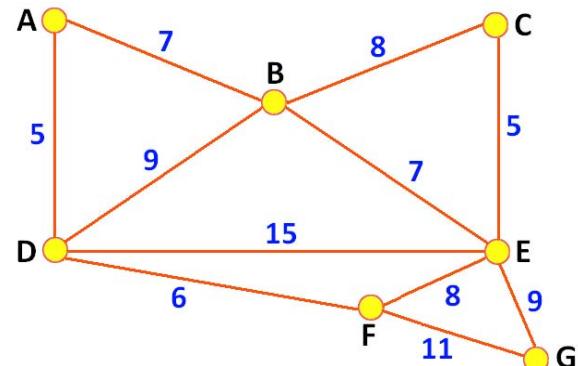
Algoritmo para encontrar un árbol recubridor mínimo en un grafo conexo y ponderado. Es decir, busca un subconjunto de aristas que, formando un árbol, incluyen todos los vértices y donde el valor de la suma de todas las aristas del árbol es mínimo.

Si el grafo fuera no conexo busca un bosque expandido mínimo (árbol expandido mínimo para cada componente conexa).

Informalmente el algoritmo se puede describir:

- Se crea un bosque (un conjunto de árboles) donde cada vértice del grafo es un árbol separado.
- Se crea un conjunto C que contenga a todas las aristas del grafo.
- Mientras el conjunto C sea no vacío:
 - ◆ Se elimina la arista de peso mínimo de C
 - ◆ Si la arista conecta dos árboles diferentes se añade al bosque, combinando los dos árboles en un solo árbol. En caso contrario, se desecha la arista.
 - ◆ El algoritmo finaliza cuando el bosque tiene un solo componente, que forma un árbol de expansión mínima del grafo.

KRUSKAL: Calcular el peso que transporta el árbol de expansión mínima del grafo.



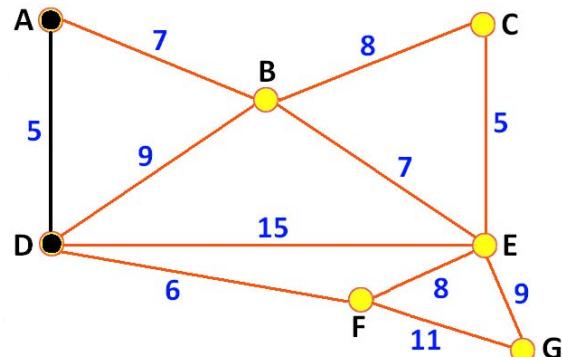
Inicialmente el grafo ponderado no es un árbol (para serlo se requiere que no hubiera ciclos y en este caso sí hay).

Comienza el proceso eligiendo las aristas con menor peso. En este caso, las aristas más cortas, con un peso de 5 , son AD y CE.

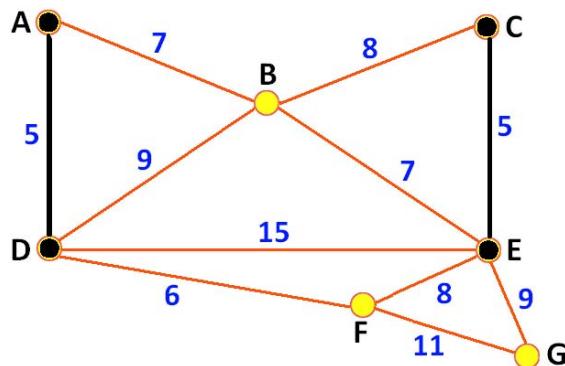
Se elige arbitrariamente la arista AD y se marca.

Ahora es CE la arista más pequeña que no forma ciclos, con un peso de 5.

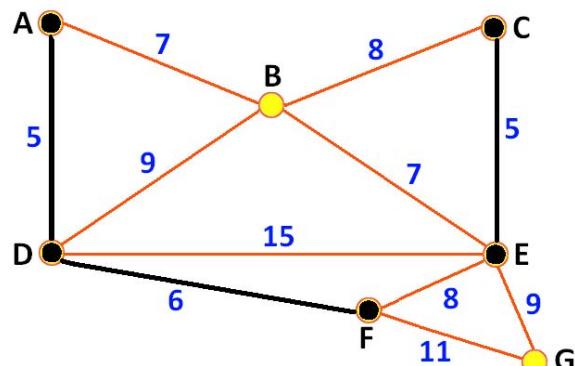
Se marca la arista CE como segunda arista.



La siguiente arista con menor peso que no forma ciclos es DF con un peso de 6

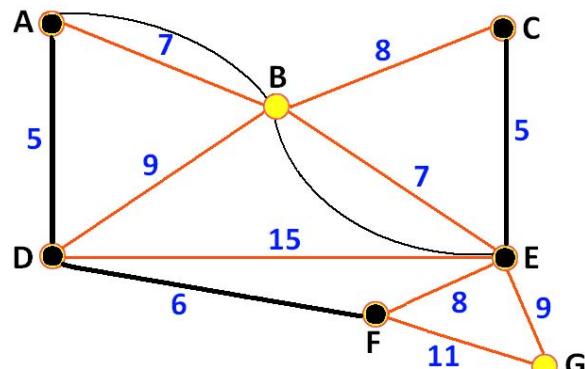


Se marca la arista DF como tercera arista.



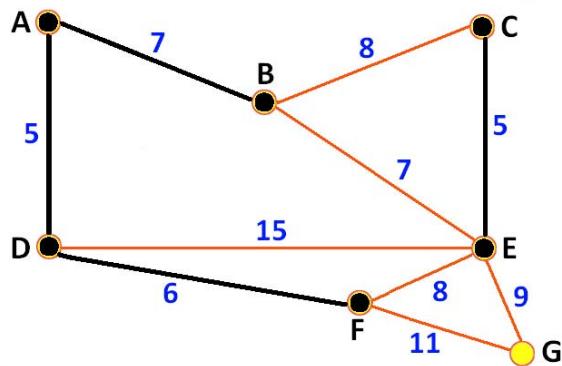
Las siguientes aristas más pequeñas que no forman ciclos, con un peso de 7, son AB y BE.

Se elige arbitrariamente AB y se marca.



Se elimina la arista BD porque formaría un ciclo ABD si fuera elegida.

El algoritmo continua, se marca la arista más pequeña BE, que no forma ciclos, con un peso de 7.

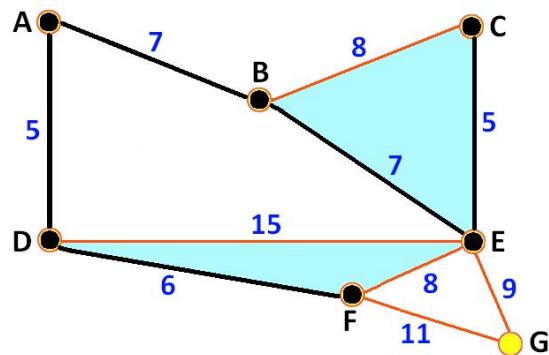


Se elimina la arista BC porque formaría un ciclo con BCE.

Por el mismo criterio se eliminan otras dos aristas:

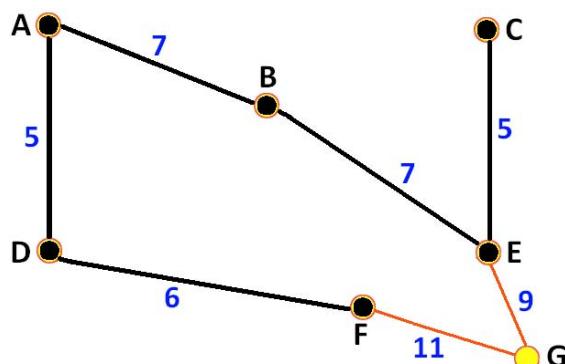
La arista DE porque formaría ciclo con DEBA.

La arista FE porque formaría ciclo con FEBAD.

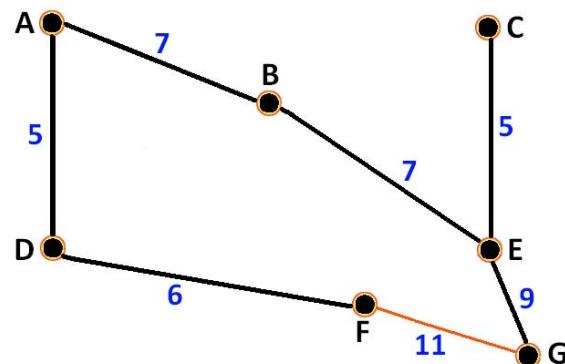


El algoritmo finaliza eligiendo la arista más pequeña que no forma ciclo.

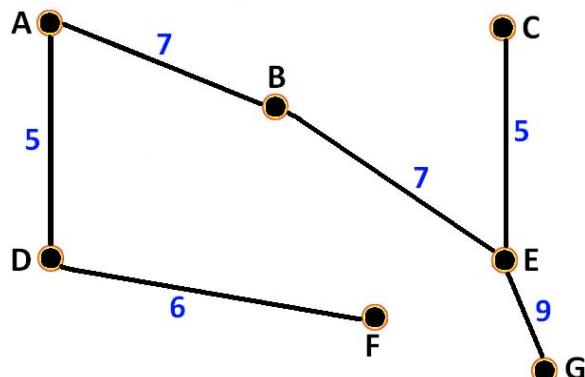
Se elige la arista EG con un peso de 9.



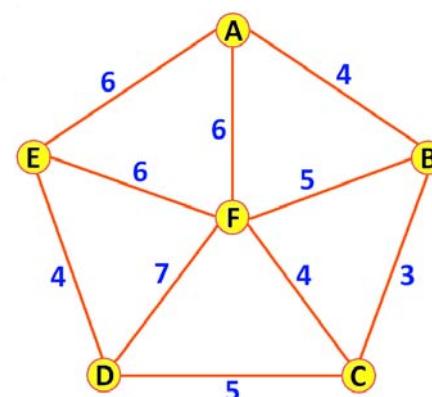
Se elimina la arista FG porque formaría ciclo.



El árbol de expansión mínima tiene un peso de 39.



- KRUSKAL: Construir un árbol generador del grafo ponderado adjunto utilizando el algoritmo de Kruskal.

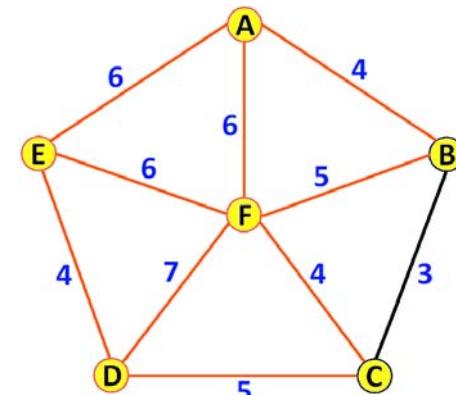


El algoritmo de Kruskal comienza ordenando las aristas por su peso de menor a mayor. Posteriormente, en ese orden, se van añadiendo aristas al árbol si no forman ciclo con las previamente añadidas.

El grafo ponderado de partida no es un árbol, ya que para serlo requiere que no tuviera ciclos.

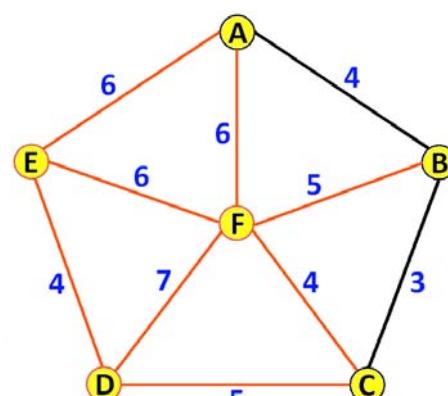
BC es la arista más corta, con peso 3.

Se marca la arista BC.



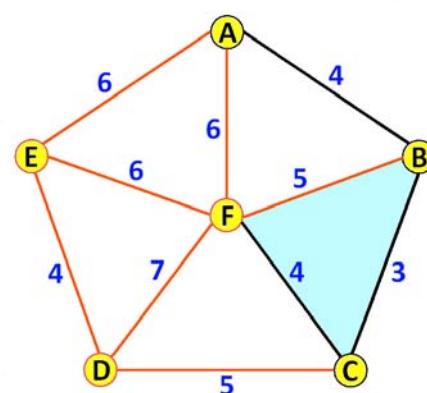
Las aristas AB, CF y DE son las aristas más cortas, con peso 4. Se elige arbitrariamente la arista AB.

En el siguiente paso, quedan como aristas más cortas CF y DE.



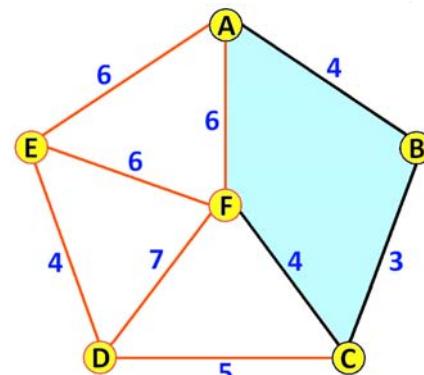
Se elige arbitrariamente la arista CF.

El proceso continúa, la arista FB debe eliminarse porque formaría el ciclo BCF.



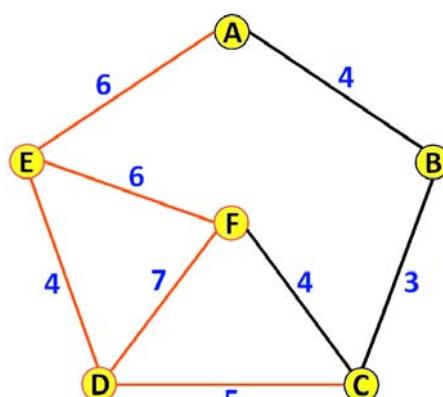
Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

Antes de continuar con el algoritmo se tiene que eliminar la arista AF porque formaría el ciclo ABCF.



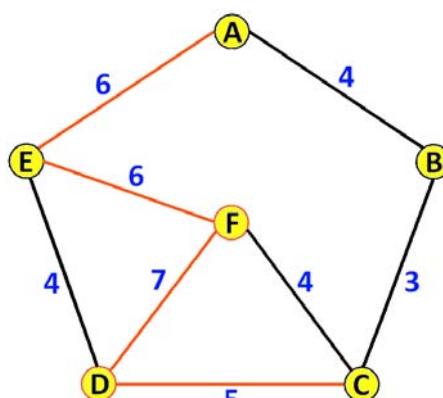
La siguiente arista más corta es DE, con peso 4.

Se marca DE.



La siguiente arista más corta es CD, con peso 5.

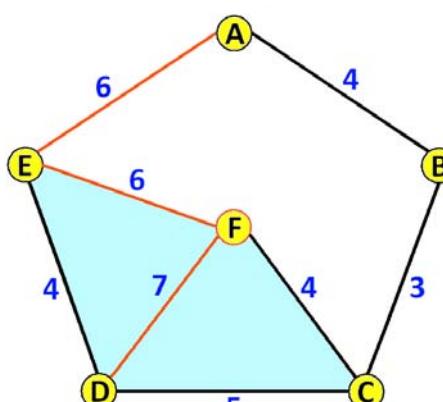
Se marca DE.



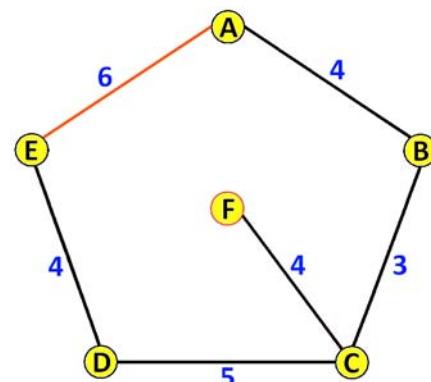
Al marcar la arista DE desaparece la arista DF y la arista EF.

La arista DF formaría el ciclo CDF.

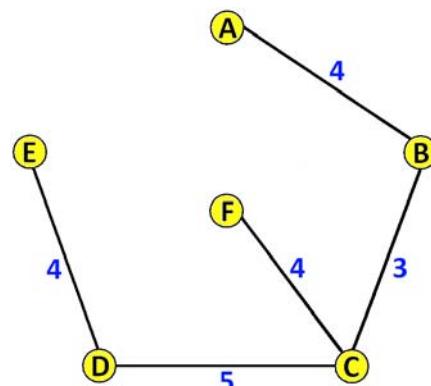
La arista EF formaría el ciclo CDEF.



Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día



Finalmente, queda la arista AE que tiene que desaparecer porque formaría el ciclo del grafo.



Se ha encontrado el árbol de expansión mínima o árbol generador mínimo, que transporta un peso de 20.

$$V(T) = \{A, B, C, D, E, F\}$$

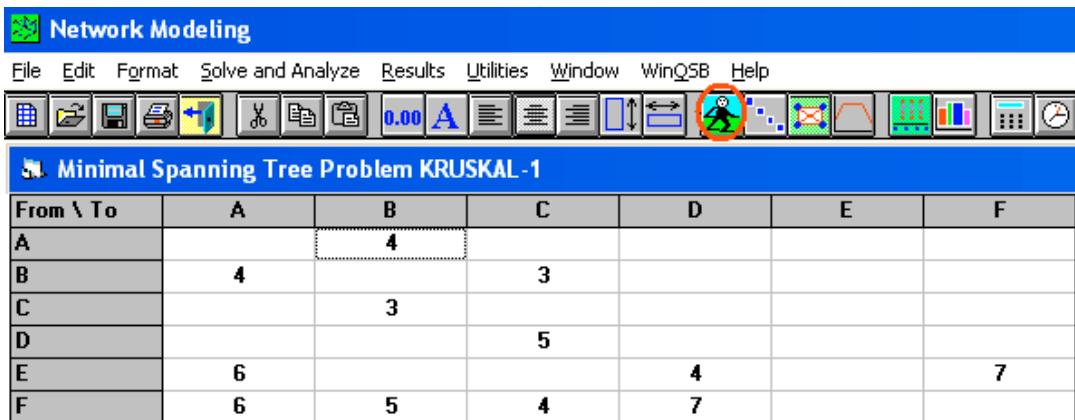


KRUSKAL: Network Modeling / Minimal Spanning Tree

NET Problem Specification

Problem Type	Objective Criterion	
<input type="radio"/> Network Flow <input type="radio"/> Transportation Problem <input type="radio"/> Assignment Problem <input type="radio"/> Shortest Path Problem <input type="radio"/> Maximal Flow Problem <input checked="" type="radio"/> Minimal Spanning Tree <input type="radio"/> Traveling Salesman Problem	<input checked="" type="radio"/> Minimization <input type="radio"/> Maximization	
Data Entry Format		
<input checked="" type="radio"/> Spreadsheet Matrix Form <input type="radio"/> Graphic Model Form <input checked="" type="checkbox"/> Symmetric Arc Coefficients <i>(i.e., both ways same cost)</i>		
Problem Title	KRUSKAL-1	
Number of Nodes	6	
OK	Cancel	Help

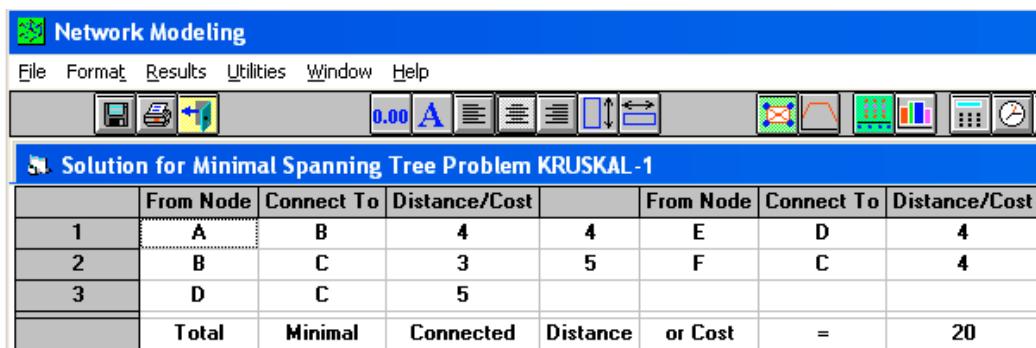
Network Modeling



File Edit Format Solve and Analyze Results Utilities Window WinQSB Help

From \ To	A	B	C	D	E	F
A		4				
B	4		3			
C		3				
D			5			
E	6			4		7
F	6	5	4	7		

Network Modeling



File Format Results Utilities Window Help

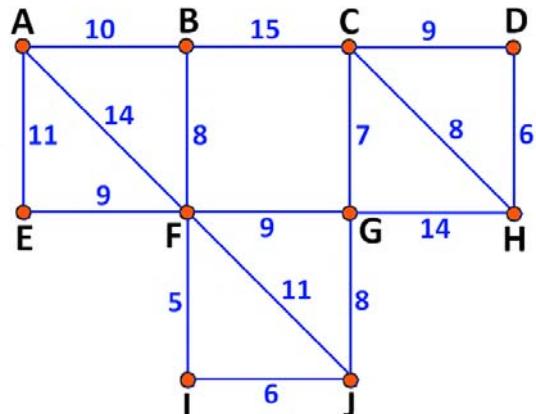
Solution for Minimal Spanning Tree Problem KRUSKAL-1

	From Node	Connect To	Distance/Cost		From Node	Connect To	Distance/Cost
1	A	B	4	4	E	D	4
2	B	C	3	5	F	C	4
3	D	C	5				
	Total	Minimal	Connected	Distance	or Cost	=	20

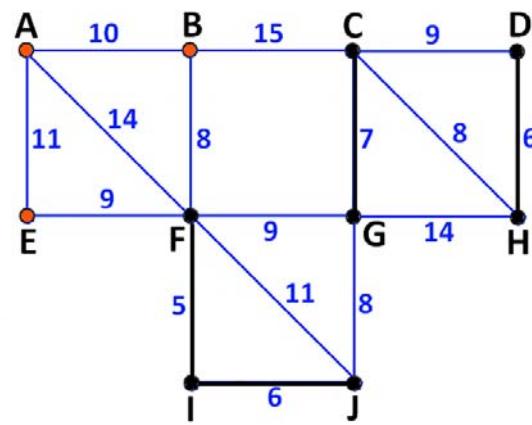
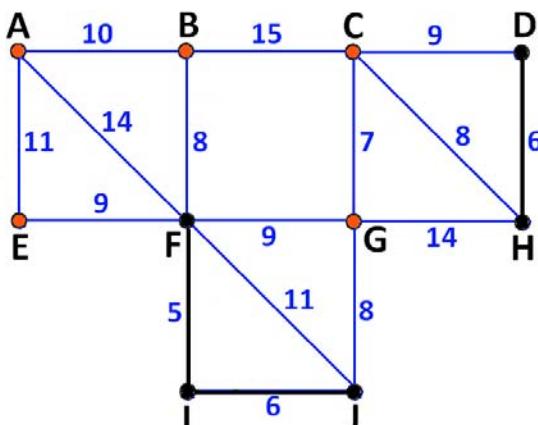
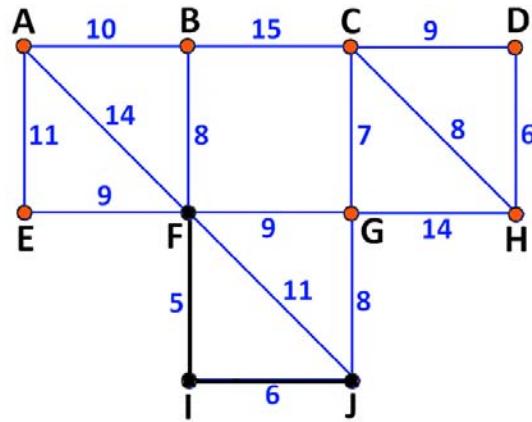
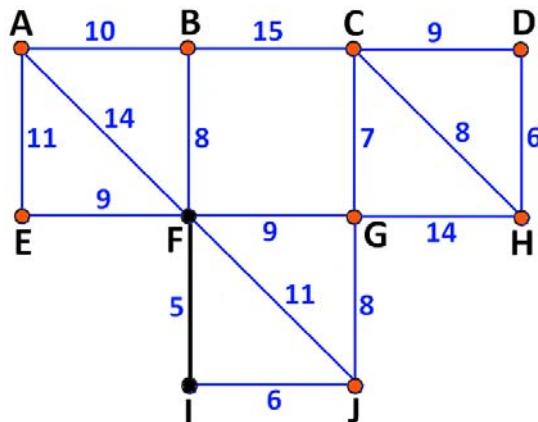
Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

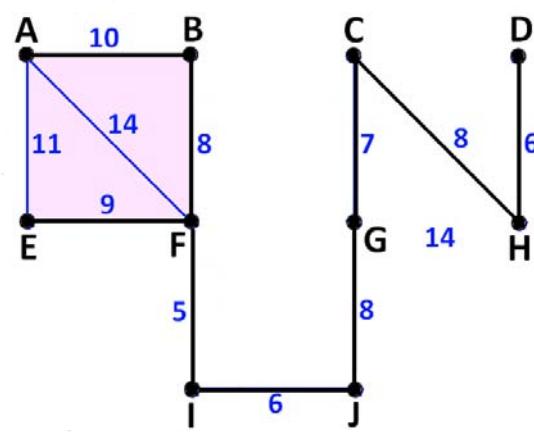
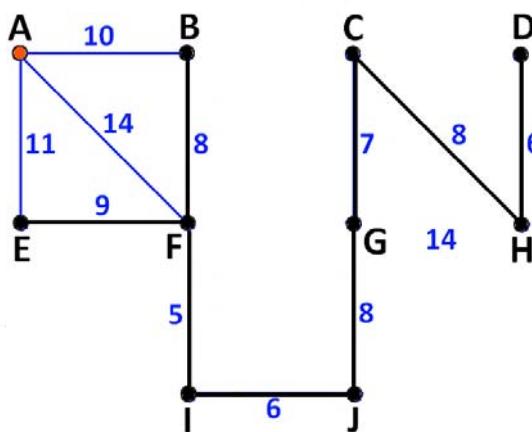
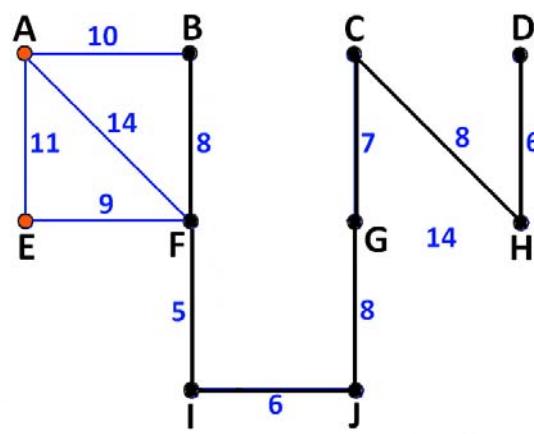
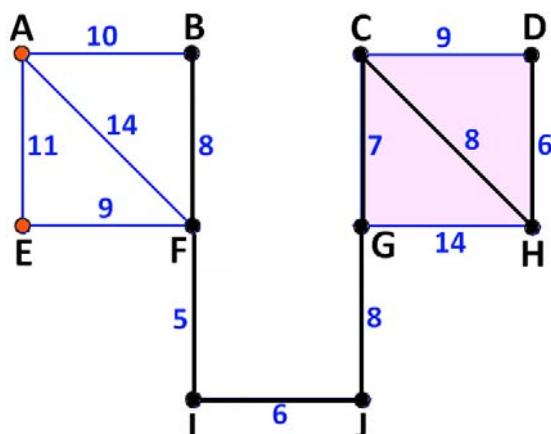
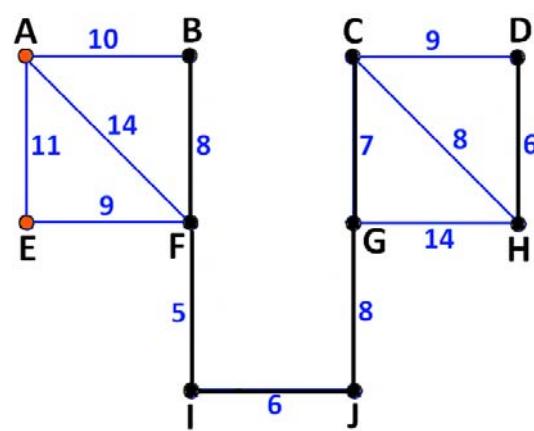
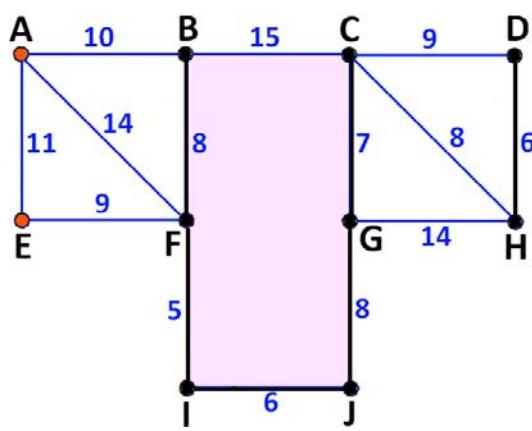
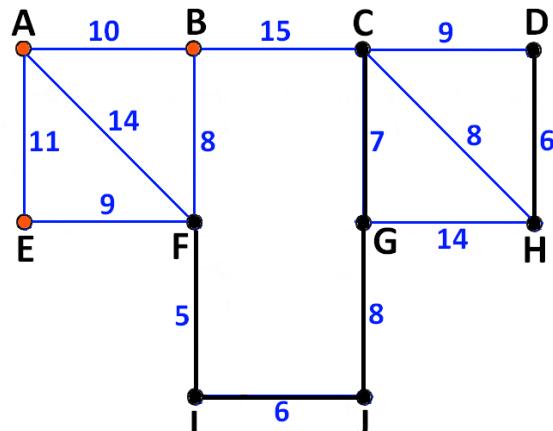
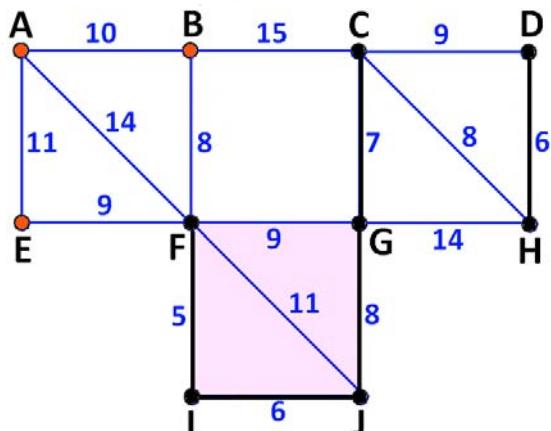
 KRUSKAL: Una empresa decide instalar un sistema de distribución que permite enviar sus artículos a diez provincias españolas, considerando la posibilidad de envío no directo. Los costes en cientos de euros se adjuntan en el grafo adjunto, ¿entre qué lugares debe instalar el canal de suministro para minimizar el coste total?

Si el coste de cada conexión se incrementase 50 euros, ¿varía la respuesta anterior?

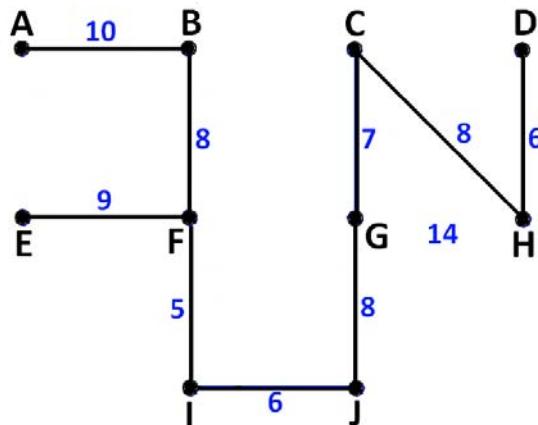


Se van eligiendo las aristas más cortas (con menor peso) y se van resaltando, cuidando que al seleccionar las aristas no formen un ciclo. Una posible secuencia sería:





Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día



El coste mínimo de la distribución es de
 $67 \times 100 = 6700$ euros.

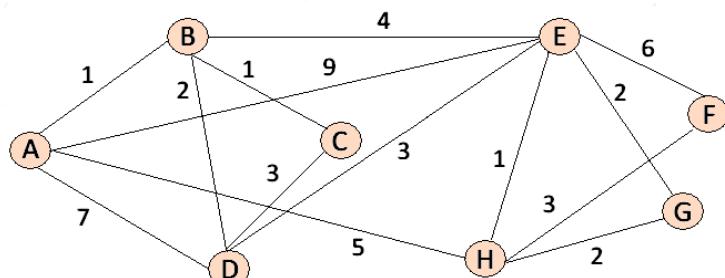
De otra parte, si el coste de la conexión se incrementa 50 euros, el trazado no varía, aunque el coste se incrementaría $9 \times 50 = 450$ euros.

El coste mínimo de la distribución sería $6700 + 450 = 7150$ euros.

Se advierte que empleando el algoritmo de Kruskal el proceso hasta encontrar el árbol expandido mínimo es tedioso, que va haciéndose más complejo a medida que aumentan los vértices del Grafo.

DIJKSTRA-KRUSKAL: La red de comunicaciones de un aeropuerto responde a la red adjunta.

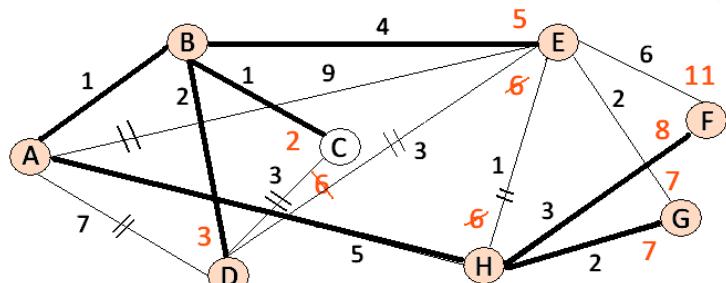
a) Encontrar la ruta más corta (peso mínimo) desde el vértice A hasta el vértice G. ¿Es única la ruta?



b) Encontrar un árbol generador de peso mínimo utilizando el algoritmo de Kruskal. ¿es euleriano del grafo?. En caso afirmativo, encontrar un circuito euleriano del grafo.

a) Aplicando el algoritmo de Dijkstra se eligen los nodos adyacentes que tienen menor peso en la arista, resultando la red que figura.

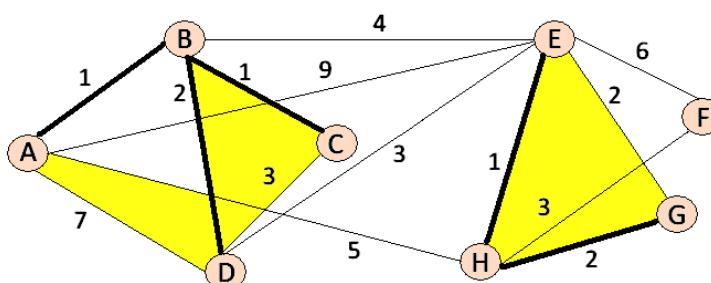
El conjunto de caminos mínimos siempre es un árbol generador.



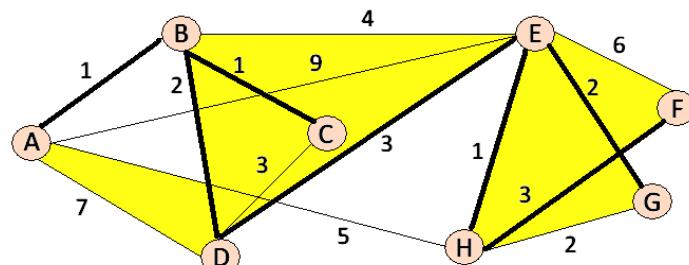
El camino más corto entre los nodos A y G tiene una longitud de 7: A - B - E - G

Hay otro camino de longitud 7: A - H - G

b) Utilizando el algoritmo de Kruskal se comienza eligiendo las aristas que tienen menor peso, es decir, las aristas más cortas, no formando ciclos, formando un conjunto que contenga a todas las aristas del grafo (el algoritmo termina cuando todos los vértices están marcados).



8 nodos → 7 aristas



El árbol generador mínimo tiene un peso de 13: $1 + 2 + 1 + 3 + 1 + 3 + 2 = 13$

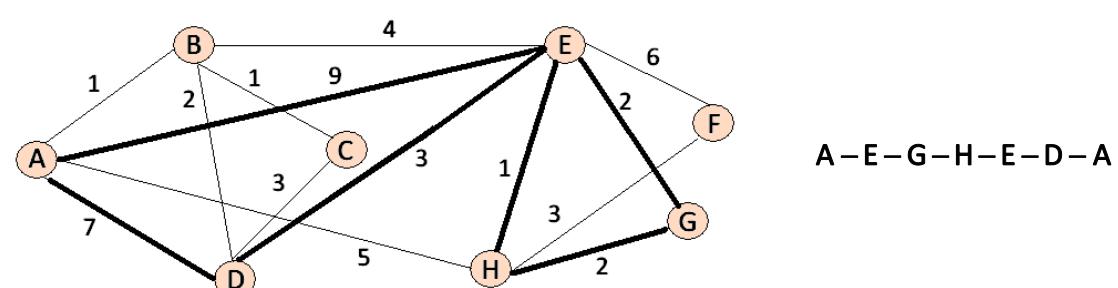
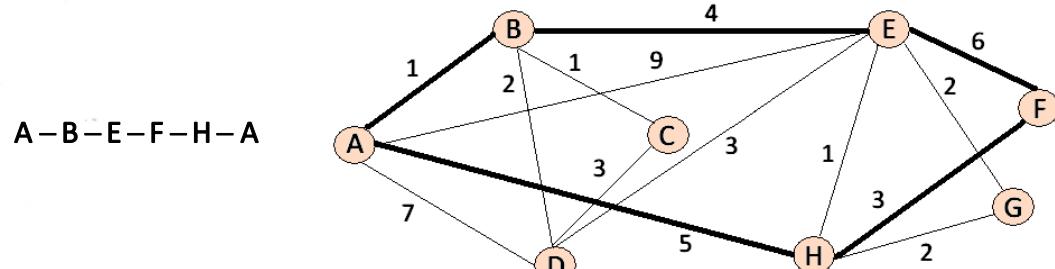
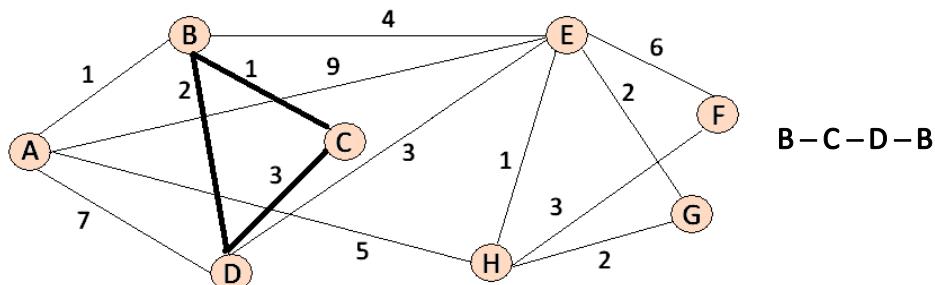
Un grafo euleriano es un camino que contiene todas las aristas y recorre cada arista una sola vez. Un grado euleriano debe ser conexo y todos los vértices deben tener grado par, a lo sumo dos vértices con grado impar.

El grafo es euleriano porque todos sus vértices son de grado par.

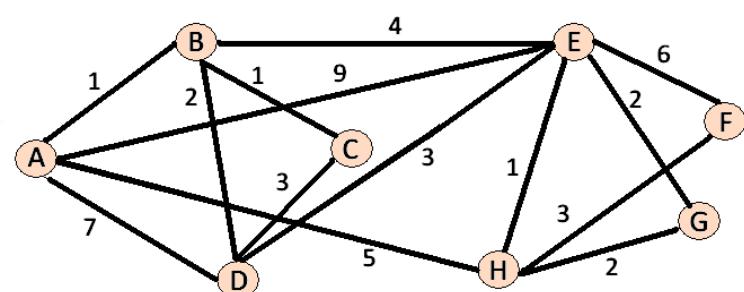
El ciclo euleriano (circuito euleriano) es un camino euleriano que comienza y termina en el mismo vértice. Todos los vértices deben ser de grado par.

Para obtener un circuito euleriano se forman ciclos cerrados:

$$\left\{ \begin{array}{l} B-C-D-B \\ A-B-E-F-H-A \\ A-E-G-H-E-D-A \end{array} \right.$$



Uniendo los ciclos, se obtiene el circuito cerrado: **A-E-G-H-E-D-A-B-C-D-B-E-F-H-A**



Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día



DIJKSTRA - RUTA MÁS CORTA: WinQSB / Network Modeling - Shortest Path Problem

NET Problem Specification

Problem Type	Objective Criterion	
<input type="radio"/> Network Flow <input type="radio"/> Transportation Problem <input type="radio"/> Assignment Problem <input checked="" type="radio"/> Shortest Path Problem <input type="radio"/> Maximal Flow Problem <input type="radio"/> Minimal Spanning Tree <input type="radio"/> Traveling Salesman Problem	<input checked="" type="radio"/> Minimization <input type="radio"/> Maximization	
Data Entry Format		
<input checked="" type="radio"/> Spreadsheet Matrix Form <input type="radio"/> Graphic Model Form <input type="checkbox"/> Symmetric Arc Coefficients <small>(i.e., both ways same cost)</small>		
Problem Title	DISTANCIA MÍNIMA	
Number of Nodes	8	
OK	Cancel	Help

Network Modeling

File Edit Format Solve and Analyze Results Utilities Window WinQSB Help

Toolbar icons: File, Edit, Format, Solve and Analyze, Results, Utilities, Window, WinQSB, Help.

Shortest Path Problem DISTANCIA MINIMA

From \ To	A	B	C	D	E	F	G	H
A		1		7				5
B			1	2	4			
C				3				
D					3			
E						6	2	1
F								3
G								
H								

Solution for Shortest Path Problem DIJKSTRA: RUTA MÁS CORTA

	From	To	Distance/Cost	Cumulative Distance/Cost
1	A	D	14	14
2	D	E	6	20
3	E	T	9	29
	From A	To T	=	29
	From A	To B	=	12
	From A	To C	=	22
	From A	To D	=	14
	From A	To E	=	20

Select Start and End Nodes

Click to select a start node

- A
- B
- C
- D
- E
- F
- G
- H

A

Click to select an end node

- A
- B
- C
- D
- E
- F
- G
- H

H

Solve

Solve and Display Steps

Cancel

Help

Network Modeling

File Format Results Utilities Window Help

0.00 A

Solution for Shortest Path Problem DISTANCIA MINIMA

	From	To	Distance/Cost	Cumulative Distance/Cost
1	A	H	5	5
	From A	To H	=	5
	From A	To B	=	1
	From A	To C	=	2
	From A	To D	=	3
	From A	To E	=	5
	From A	To F	=	11
	From A	To G	=	7

b) Para aplicar el algoritmo de Kruskal: [WinQSB / Network Modeling \(NET\) / Minimal Spanning Tree](#)

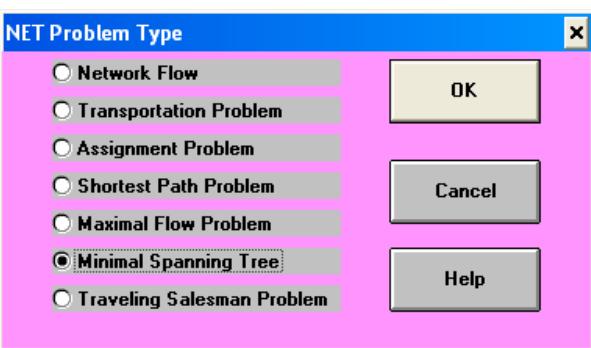
Network Modeling

File Edit Format Solve and Analyze Results Utilities Window WinQSB Help

Cut Ctrl+X
Copy Ctrl+C
Paste Ctrl+V
Clear
A : Undo
Problem Name
Node Names
Problem Type
Add a Node
Delete a Node

INSTANCIA MINIMA

From \ To	A	B	C	D	E	F	G	H
A		1		7				5
B			1	2	4			
C				3				
D					3			
E						6	2	1
F								3
G								
H								



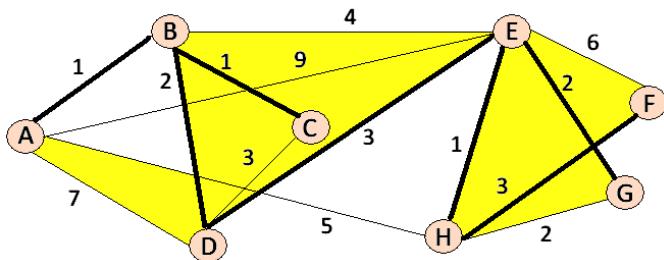
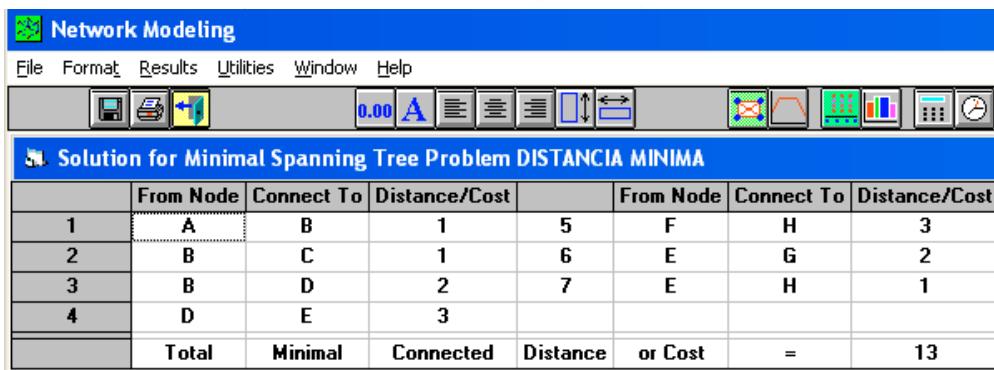
Network Modeling

File Edit Format Solve and Analyze Results Utilities Window WinQSB Help

0.00 A

Minimal Spanning Tree Problem DISTANCIA MINIMA

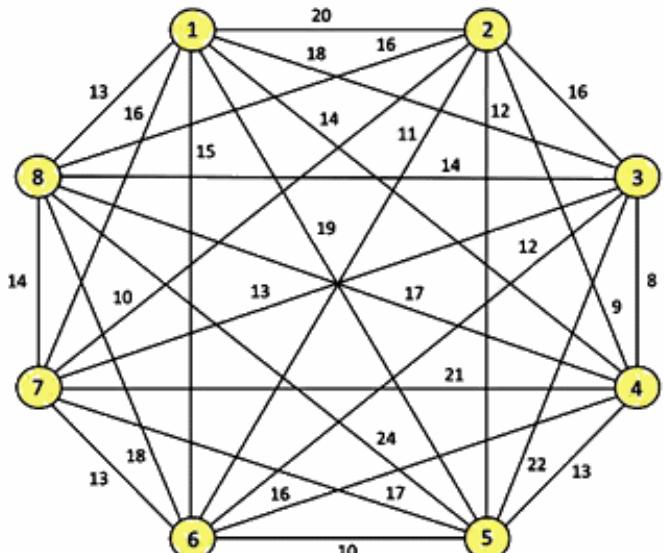
From \ To	A	B	C	D	E	F	G	H
A		1		7				5
B			1	2	4			
C				3				
D					3			
E						6	2	1
F								3
G								
H								



El árbol generador mínimo tiene un peso de 13:
 $1+2+1+3+1+3+2=13$

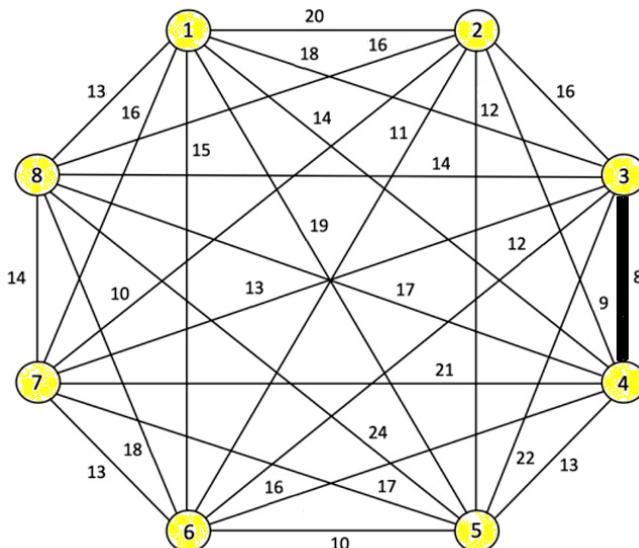
KRUSKAL (ÁRBOL EXPANSIÓN MÍNIMA)

Una empresa aeronáutica decide instalar un sistema de distribución que permita enviar contenedores a ocho provincias españolas, considerando la posibilidad de envío no directo. Los costes en cientos de euros se reflejan en la tabla adjunta. Calcular el mínimo coste.

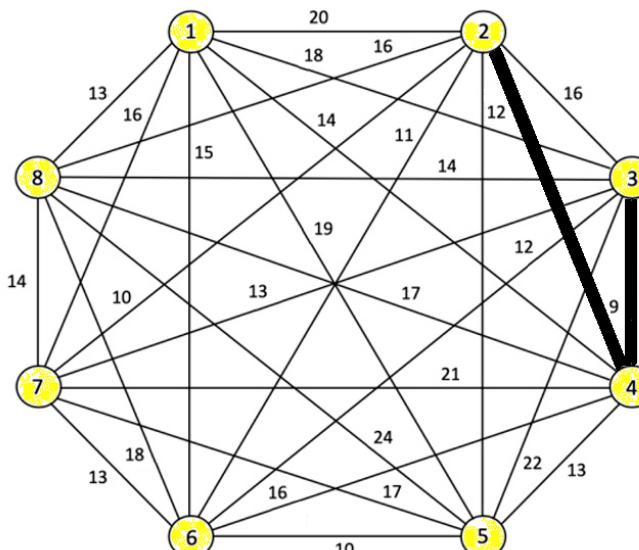


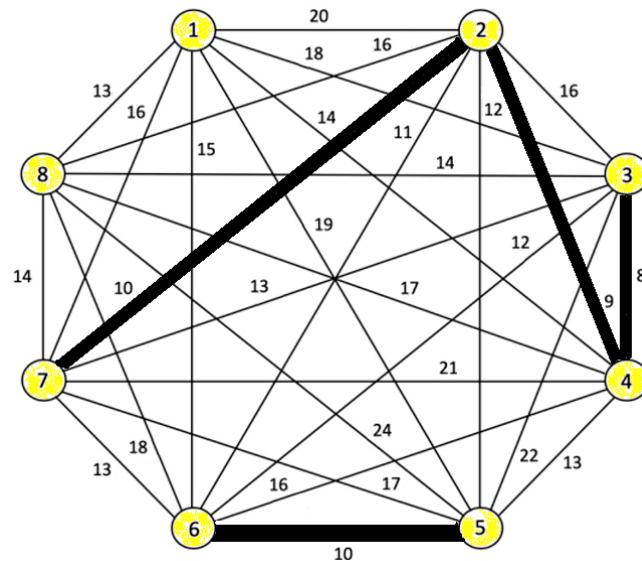
Se van seleccionando las aristas más cortas (con menor peso), ya que hay que calcular el mínimo coste. Se presta atención a las aristas marcadas para no formar un ciclo.

Se comienza seleccionando el menor valor, en este caso el 8, que une el nodo 3 y 4.

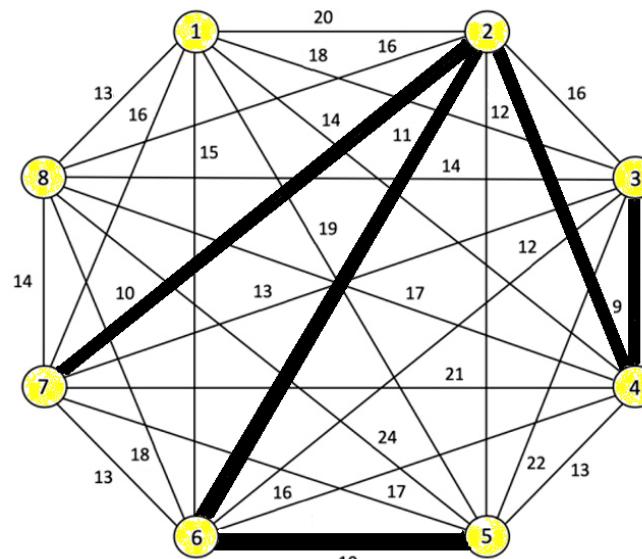


A continuación, se selecciona la arista que une los nodos 2 y 4.

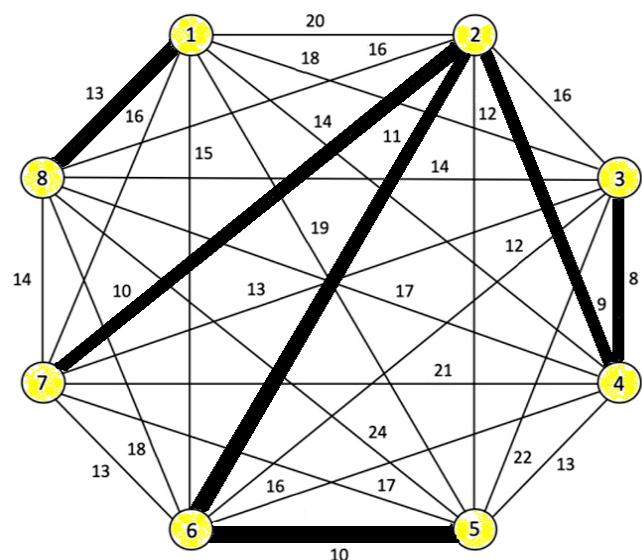




Se selecciona la arista que une los nodos 5 y 6.

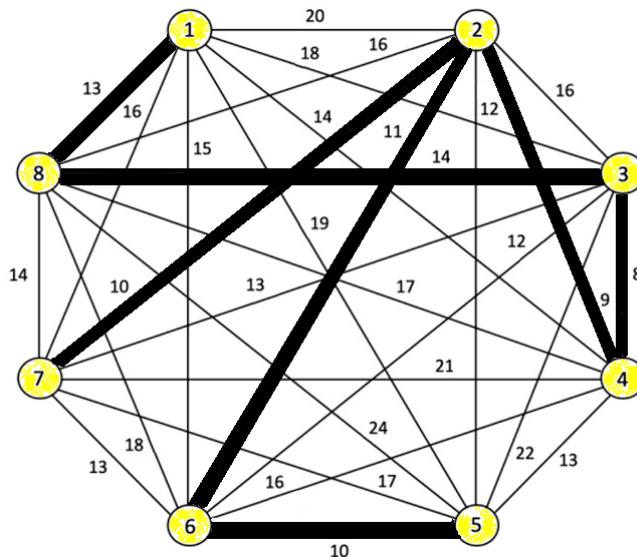


Se unen los nodos 2 y 6.

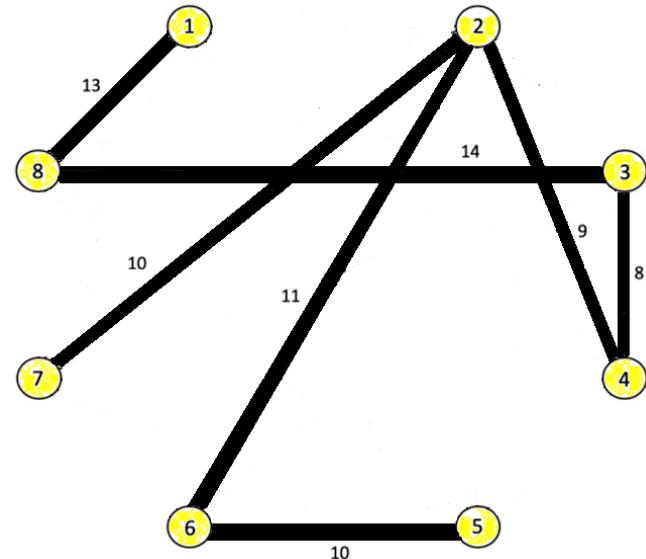


Se unen los nodos 1 y 8.

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día



Finalmente, se selecciona la
arista que une los nodos 3 y 8.



Longitud o peso mínimo de la distribución:

$$8 + 9 + 14 + 10 + 11 + 13 + 10 = 75$$

Coste mínimo de la distribución:

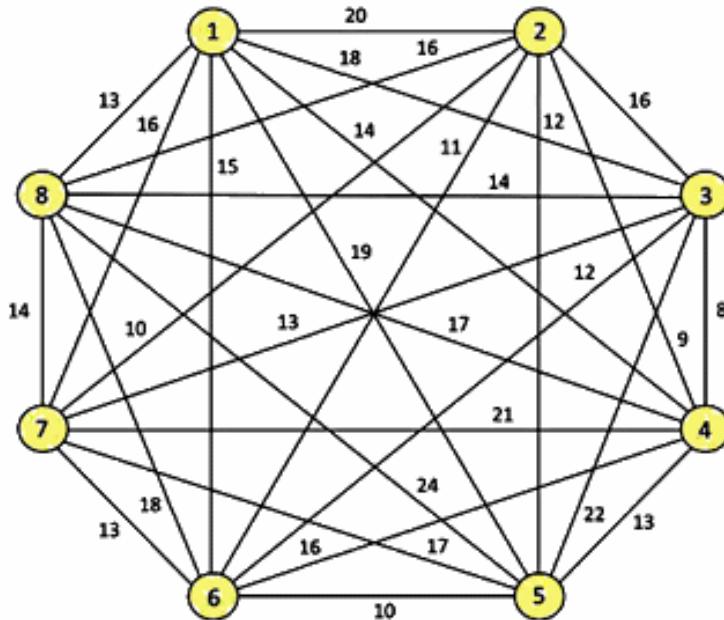
$$75 \times 100 = 7.500 \text{ euros.}$$

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

KRUSKAL (ÁRBOL EXPANSIÓN MÍNIMA)

Una empresa aeronáutica decide instalar un sistema de distribución que permita enviar contenedores a ocho provincias españolas, considerando la posibilidad de envío no directo. Los costes en cientos de euros se reflejan en la tabla adjunta. Calcular el coste mínimo.

	1	2	3	4	5	6	7	8
1	—	20	18	14	19	15	16	13
2	20	—	16	9	12	11	10	16
3	18	16	—	8	22	12	13	14
4	14	9	8	—	13	16	21	17
5	19	12	22	13	—	10	17	24
6	15	11	12	16	10	—	13	18
7	16	10	13	21	17	13	—	14
8	13	16	14	17	24	18	14	—

GRAFO ADJUNTO

Para obtener el árbol de expansión mínima se selecciona el menor elemento de la red.

En este caso es el 8, al existir dos casillas con ese número, se selecciona uno cualquiera.

distancia menor = $d_1 = 8$

	1	2	3	4	5	6	7	8
1	—	20	18	14	19	15	16	13
2	20	—	16	9	12	11	10	16
3	18	16	—	8	22	12	13	14
4	14	9	8	—	13	16	21	17
5	19	12	22	13	—	10	17	24
6	15	11	12	16	10	—	13	18
7	16	10	13	21	17	13	—	14
8	13	16	14	17	24	18	14	—

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

Se tachan las casillas (3, 4) y (4, 3) que es donde se encuentra ese mínimo.

	1	2	3	4	5	6	7	8
1	—	20	18	14	19	15	16	13
2	20	—	16	9	12	11	10	16
3	18	16	—	22	12	13	14	
4	14	9	—	13	16	21	17	
5	19	12	22	13	—	10	17	24
6	15	11	12	16	16	10	—	13
7	16	10	13	21	17	13	—	14
8	13	16	14	17	24	18	14	—

1º 2º

	1	2	3	4	5	6	7	8
1	—	20	18	14	19	15	16	13
2	20	—	16	9	12	11	10	16
3	18	16	—	22	12	13	14	
4	14	9	—	13	16	21	17	
5	19	12	22	13	—	10	17	24
6	15	11	12	16	10	—	13	18
7	16	10	13	21	17	13	—	14
8	13	16	14	17	24	18	14	—

1º 2º

Se selecciona el menor elemento de las columnas no marcadas, el elemento 9 de la casilla (2, 4).

Se tachan las casillas (2, 4) y (4, 2) que es donde se encuentra ese mínimo.

Asimismo, se tachan las casillas (3, 2) y (2, 3) para evitar ciclos (elemento 16).

distancia menor = $d_2 = 9$

Marcando la columna 2 que es donde se ha eliminado la casilla (2, 3).

	1	2	3	4	5	6	7	8
1	—	20	18	14	19	15	16	13
2	20	—	—	—	12	11	10	16
3	18	—	—	—	22	12	13	14
4	14	—	—	—	13	16	21	17
5	19	12	22	13	—	10	17	24
6	15	11	12	16	10	—	13	18
7	16	10	13	21	17	13	—	14
8	13	16	14	17	24	18	14	—

3º 1º 2º

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

Se selecciona el menor elemento de las columnas no marcadas, que es el elemento 10 de la casilla (7, 2).

Se tachan las casillas (7, 2), (7, 3) y (7, 4), así como sus casillas simétricas: (2, 7), (3, 7) y (4, 7) para evitar ciclos.

distancia menor = $d_3 = 10$

	1	2	3	4	5	6	7	8
1	—	20	18	14	19	15	16	13
2	20	—			12	11	10	16
3	18		—		22	12	13	14
4	14			—	13	16	21	17
5	19	12	22	13	—	10	17	24
6	15	11	12	16	10	—	13	18
7	16	10	13	21	17	13	—	14
8	13	16	14	17	24	18	14	—

3º 1º 2º

Marcando la columna 7 que es donde se ha eliminado la casilla (7, 2).

	1	2	3	4	5	6	7	8
1	—	20	18	14	19	15	16	13
2	20	—			12	11		16
3	18		—		22	12		14
4	14			—	13	16		17
5	19	12	22	13	—	10	17	24
6	15	11	12	16	10	—	13	18
7	16				17	13	—	14
8	13	16	14	17	24	18	14	—

3º 1º 2º 4º

Se selecciona el menor elemento de las columnas no marcadas, que es el elemento 10 de la casilla (5, 6).

Se tachan las casillas (5, 6), (5, 2), (5, 3), (5, 4) y (5, 7), así como sus casillas simétricas: (6, 5), (2, 5), (3, 5), (4, 5) y (7, 5) para evitar ciclos.

distancia menor = $d_4 = 10$

	1	2	3	4	5	6	7	8
1	—	20	18	14	19	15	16	13
2	20	—			12	11		16
3	18		—		22	12		14
4	14			—	13	16		17
5	19	22	22	23	—	10	17	24
6	15	11	12	16	10	—	13	18
7	16				17	13	—	14
8	13	16	14	17	24	18	14	—

3º 1º 2º 4º

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

	1	2	3	4	5	6	7	8
1	-	20	18	14	19	15	16	13
2	20	-				11		16
3	18		-			12		14
4	14			-		16		17
5	19				-			24
6	15	11	12	16		-	13	18
7	16				13	-	14	
8	13	16	14	17	24	18	14	-

3º 1º 2º 5º 4º

Marcando la columna 5 que es donde se ha seleccionado la casilla (5, 6).

	1	2	3	4	5	6	7	8
1	-	20	18	14	19	15	16	13
2	20	-				11		16
3	18		-			12		14
4	14			-		16		17
5	19				-			24
6	15	11	12	16		-	13	18
7	16				13	-	14	
8	13	16	14	17	24	18	14	-

3º 1º 2º 5º 4º

Se selecciona el menor elemento de las columnas no marcadas, que es el elemento 11 de la casilla (6, 2).

Se tachan las casillas (6, 2), (6, 3), (6, 4) y (6, 7), así como sus casillas simétricas: (2, 6), (3, 6), (4, 6) y (7, 6) para evitar ciclos.

distancia menor = $d_5 = 11$

	1	2	3	4	5	6	7	8
1	-	20	18	14	19	15	16	13
2	20	-					16	
3	18		-					14
4	14			-				17
5	19				-			24
6	15					-		18
7	16					-		14
8	13	16	14	17	24	18	14	-

3º 1º 2º 5º 6º 4º

Marcando la columna 6 que es donde se ha seleccionado la casilla (6, 2).

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

Se selecciona el menor elemento de las columnas no marcadas, que es el elemento 13 de la casilla (8, 1).

Se tachan las casillas (8, 1), (8, 2), (8, 3), (8, 4), (8, 5), (8, 6) y (8, 7), así como sus casillas simétricas: (1, 8), (2, 8), (3, 8), (4, 8), (5, 8), (6, 8) y (7, 8) para evitar ciclos.

distancia menor = $d_6 = 13$

	1	2	3	4	5	6	7	8
1	—	20	18	14	19	15	16	13
2	20	—						16
3	18		—					14
4	14			—				17
5	19				—			24
6	15					—		18
7	16						—	14
8	13	16	14	17	24	18	14	—
	3º	1º	2º	5º	6º	4º	7º	

Marcando la columna 8 que es donde se ha seleccionado la casilla (8, 1).

	1	2	3	4	5	6	7	8
1	—	20	18	14	19	15	16	
2	20	—						
3	18		—					
4	14			—				
5	19				—			
6	15					—		
7	16						—	
8								—
	3º	1º	2º	5º	6º	4º	7º	

Se selecciona el menor elemento de las columnas no marcadas, que es el elemento 14 de la casilla (1, 4).

Se tachan todas las casillas de la columna 1, al igual que sus simétricas.

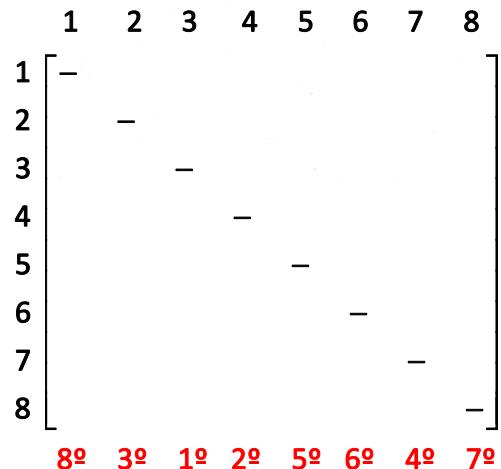
El algoritmo de Kruskal ha finalizado al tener todas las columnas marcadas.

distancia menor = $d_7 = 14$

	1	2	3	4	5	6	7	8
1	—	20	18	14	19	15	16	
2	20	—						
3	18		—					
4	14			—				
5	19				—			
6	15					—		
7	16						—	
8								—
	3º	1º	2º	5º	6º	4º	7º	

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

Marcando la columna 1 como la última,
 que es donde se ha seleccionado la
 casilla (1, 4).



Longitud o peso mínimo de la distribución: $8 + 9 + 10 + 10 + 11 + 13 + 14 = 75$

Coste mínimo de la distribución: $75 \times 100 = 7.500$ euros

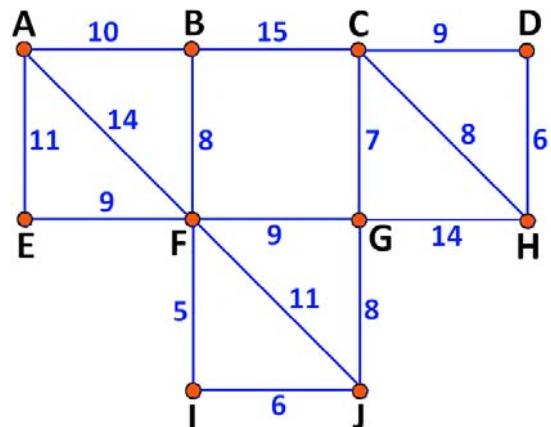
ALGORITMO DE SOLLIN: ÁRBOL DE EXPANSIÓN MÍNIMA

El algoritmo de Boruvka en la década de 1960 fue descubierto por Sollin, científico en computación, y frecuentemente es conocido como el **algoritmo de SOLLIN**, especialmente en computación paralela donde muchas instrucciones se ejecutan simultáneamente, operando sobre el principio de que problemas grandes, a menudo se pueden dividir en problemas más pequeños, que luego son resueltos simultáneamente.

El algoritmo de Sollin procede de la siguiente forma:

1. Se elige el elemento de menor valor, en caso de que existan varios se elige uno cualquiera. Se marca el elemento y se elimina la columna donde se encuentre el valor marcado y se marca la fila como primera fila marcada.
2. Del conjunto de elementos de las filas marcadas y que no han sido eliminados, se elige el valor mínimo y se marca. Se elimina la columna donde se encuentre el valor marcado y se marca la fila como segunda fila marcada.
3. Se reitera el proceso desde el punto (2) hasta que no se puedan seleccionar más elementos.

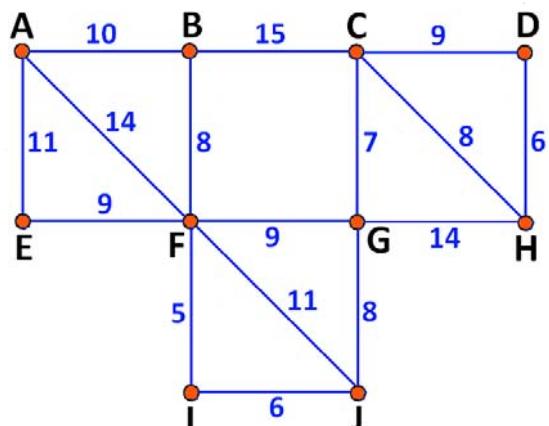
SOLLIN: Mediante un proceso sistemático, desde el principio de que a menudo problemas complejos se pueden dividir en problemas más pequeños, ejecutándolos después simultáneamente, posibilita el cálculo del árbol expandido mínimo de una forma más sencilla.



Al iniciar el algoritmo se elige el elemento de menor valor.

En este caso hay dos elementos con valor 5, se selecciona arbitrariamente uno de ellos, por ejemplo el 5 de la columna F y se marca.

Se elimina la columna F y se marca la fila F como la primera.



	A	B	C	D	E	F	G	H	I	J
	10				11	14				
	10		15			8				
				9			7	8		
				9				6		
	11					9				
1 →	14	8			9		9		5	11
			7			9		14		8
			8	6			14			
I						5 ¹				6
J						11	8		6	

De los elementos de la fila $\xrightarrow{1}$ F marcada se elige el elemento de menor valor que es 5 y se marca. Se elimina la columna I y se marca la fila I como la segunda.

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

	A	B	C	D	E	F	G	H	I	J
A		10			11	14				
B	10		15			8				
C		15		9			7	8		
D			9					6		
E	11					9				
1 →	F	14	8		9		9		5 ²	11
G			7			9		14		8
H			8	6			14			
2 →	I					5 ¹				6
J						11	8		6	

De las filas marcadas hasta este momento (F, I) se elige el elemento con menor valor y no marcado anteriormente. En la fila I está el elemento 6 y se marca, eliminando la columna J donde se encuentra, marcando la fila J como la tercera.

	A	B	C	D	E	F	G	H	I	J
A		10			11	14				
B	10		15			8				
C		15		9			7	8		
D			9					6		
E	11					9				
1 →	F	14	8		9		9		5 ²	11
G			7			9		14		8
H			8	6			14			
2 →	I					5 ¹				6 ³
3 →	J					11	8		6	

De las filas marcadas hasta ahora (F, I, J) se elige el elemento con menor valor y no marcado anteriormente, que es un 8. Se elige arbitrariamente el 8 de la columna G, eliminando la columna G y marcando la fila G como la cuarta.

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

	A	B	C	D	E	F	G	H	I	J
	A	10			11	14				
	B	10		15		8				
	C		15		9			7	8	
	D			9					6	
	E	11					9			
<u>1</u> →	F	14	8			9		9		<u>5²</u>
<u>4</u> →	G			7			9		14	
	H			8	6			14		
<u>2</u> →	I						<u>5¹</u>			<u>6³</u>
<u>3</u> →	J						11	<u>8⁴</u>		6

Sigue el proceso, de las filas marcadas (F, I, J, G) se elige el elemento con menor valor y no marcado anteriormente. En la fila G está el elemento 7 y se marca, eliminando la columna C y marcando la fila C como la quinta.

	A	B	C	D	E	F	G	H	I	J
	A	10			11	14				
	B	10		15		8				
<u>5</u> →	C		15		9			7	8	
	D			9					6	
	E	11					9			
<u>1</u> →	F	14	8			9		9		<u>5²</u>
<u>4</u> →	G			<u>7⁵</u>			9		14	
	H			8	6			14		
<u>2</u> →	I						<u>5¹</u>			<u>6³</u>
<u>3</u> →	J						11	<u>8⁴</u>		6

De las filas marcadas (F, I, J, G, C) se elige el elemento con menor valor y no marcado anteriormente. El elemento con menor valor es el 8, que se encuentra en la fila C y en la fila F, se elige arbitrariamente el 8 de la columna B, eliminando la columna y marcando la fila B como la sexta.

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

	A	B	C	D	E	F	G	H	I	J
	A		10			11	14			
<u>6</u> →	B	10		15			8			
<u>5</u> →	C		15		9			7	8	
	D			9					6	
	E	11					9			
<u>1</u> →	F	14	^{8⁶}			9		9		^{5²} 11
<u>4</u> →	G			^{7⁵}			9		14	8
	H			8	6			14		
<u>2</u> →	I					^{5¹}				^{6³}
<u>3</u> →	J					11	^{8⁴}		6	

De las filas marcadas (F, I, J, G, C) se elige el elemento con menor valor y no marcado anteriormente. El elemento con menor valor es el 8 que se encuentra en la columna H, se elimina la columna y se marca la fila H como la séptima.

	A	B	C	D	E	F	G	H	I	J
	A		10			11	14			
<u>6</u> →	B	10		15			8			
<u>5</u> →	C		15		9			7	^{8⁷}	
	D			9					6	
	E	11					9			
<u>1</u> →	F	14	^{8⁶}			9		9		^{5²} 11
<u>4</u> →	G			^{7⁵}			9		14	8
<u>7</u> →	H			8	6			14		
<u>2</u> →	I					^{5¹}				^{6³}
<u>3</u> →	J					11	^{8⁴}		6	

De las filas marcadas (F, I, J, G, C, H) se elige el elemento con menor valor y no marcado anteriormente. El elemento con menor valor es el 6 que se encuentra en la columna D, se elimina la columna y se marca la fila D como la octava.

	A	B	C	D	E	F	G	H	I	J
	A	10			11	14				
<u>6</u> →	B	10		15		8				
<u>5</u> →	C		15		9		7	<u>8⁷</u>		
<u>8</u> →	D			9				6		
	E	11					9			
<u>1</u> →	F	14	<u>8⁶</u>			9		9	<u>5²</u>	11
<u>4</u> →	G			<u>7⁵</u>			9		14	8
<u>7</u> →	H			8	<u>6⁸</u>			14		
<u>2</u> →	I					<u>5¹</u>				<u>6³</u>
<u>3</u> →	J					11	<u>8⁴</u>		6	

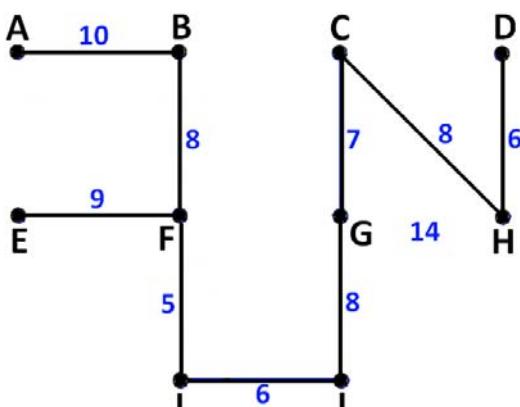
De las filas marcadas (F, I, J, G, C, H, D) se elige el elemento con menor valor y no marcado anteriormente. El elemento con menor valor es el 9 que se encuentra en la columna E, se elimina la columna y se marca la fila E como la novena.

	A	B	C	D	E	F	G	H	I	J
	A	10			11	14				
<u>6</u> →	B	10		15		8				
<u>5</u> →	C		15		9		7	<u>8⁷</u>		
<u>8</u> →	D			9				6		
<u>9</u> →	E	11					9			
<u>1</u> →	F	14	<u>8⁶</u>			<u>9⁹</u>		9	<u>5²</u>	11
<u>4</u> →	G			<u>7⁵</u>			9		14	8
<u>7</u> →	H			8	<u>6⁸</u>			14		
<u>2</u> →	I					<u>5¹</u>				<u>6³</u>
<u>3</u> →	J					11	<u>8⁴</u>		6	

Finalmente, el proceso finaliza marcando el menor valor de la columna A que es el 10, se elimina la columna y se marca la fila A como la décima.

	A	B	C	D	E	F	G	H	I	J
10 →	A		10		11	14				
6 →	B	10 ¹⁰		15		8				
5 →	C		15		9		7	8 ⁷		
8 →	D			9				6		
9 →	E	11				9				
1 →	F	14	8 ⁶			9 ⁹	9		5 ²	11
4 →	G			7 ⁵			9	14		8
7 →	H			8	6 ⁸		14			
2 →	I					5 ¹				6 ³
3 →	J					11	8 ⁴	6		

En un diagrama:

Coste mínimo de la distribución: $67 \times 100 = 6700$ euros.

De otra parte, si el coste de la conexión se incrementa 50 euros, el trazado no varía, aunque el coste se incrementaría $9 \times 50 = 450$ euros. El coste mínimo de la distribución sería $6700 + 450 = 7150$ euros.



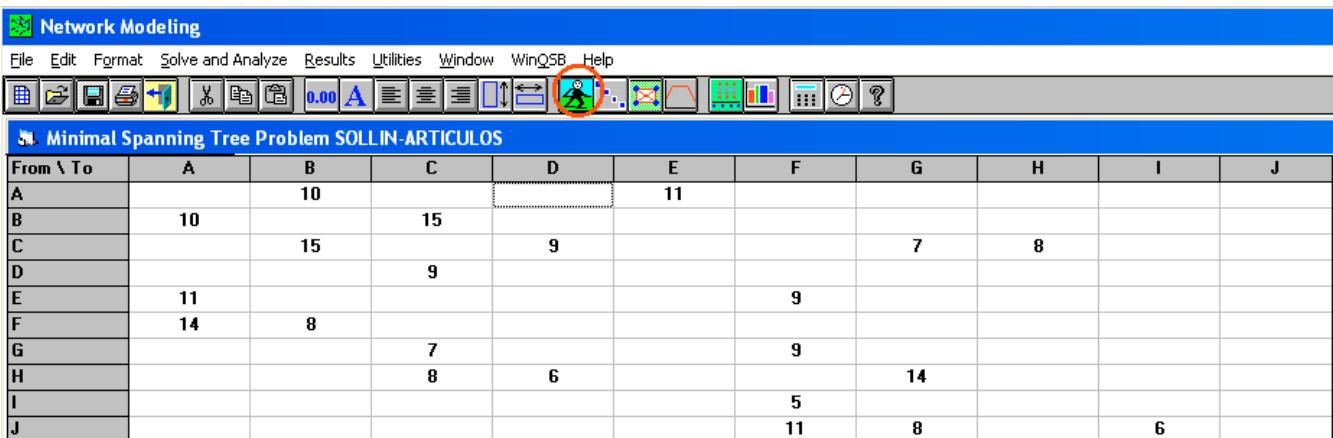
SOLLIN: Network Modeling / Minimal Spanning Tree

NET Problem Specification

Problem Type	Objective Criterion	
<input type="radio"/> Network Flow <input type="radio"/> Transportation Problem <input type="radio"/> Assignment Problem <input type="radio"/> Shortest Path Problem <input type="radio"/> Maximal Flow Problem <input checked="" type="radio"/> Minimal Spanning Tree <input type="radio"/> Traveling Salesman Problem	<input checked="" type="radio"/> Minimization <input type="radio"/> Maximization	
Data Entry Format		
<input checked="" type="radio"/> Spreadsheet Matrix Form <input type="radio"/> Graphic Model Form <input checked="" type="checkbox"/> Symmetric Arc Coefficients <small>[i.e., both ways same cost]</small>		
Problem Title	SOLLIN-1	
Number of Nodes	10	
OK	Cancel	Help

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

Network Modeling

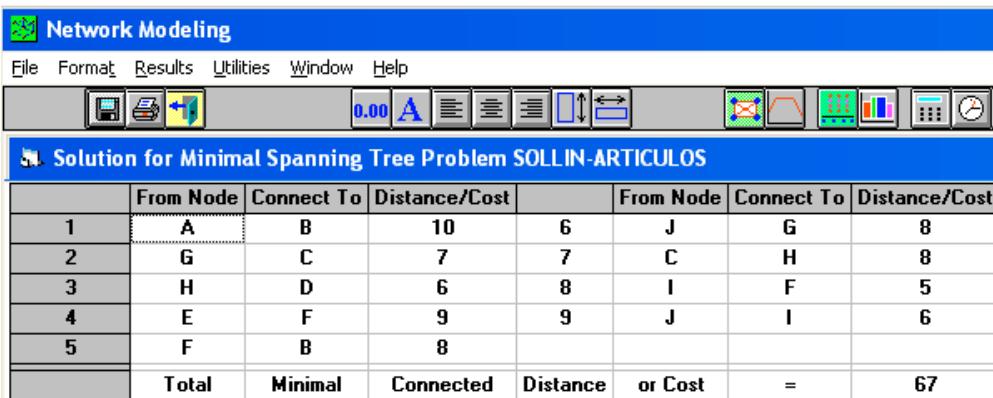


File Edit Format Solve and Analyze Results Utilities Window WinQSB Help

Minimal Spanning Tree Problem SOLLIN-ARTICULOS

From \ To	A	B	C	D	E	F	G	H	I	J
A		10				11				
B	10		15							
C		15		9			7	8		
D			9							
E	11					9				
F	14	8				9				
G			7							
H			8	6			14			
I						5				
J						11	8			6

Network Modeling

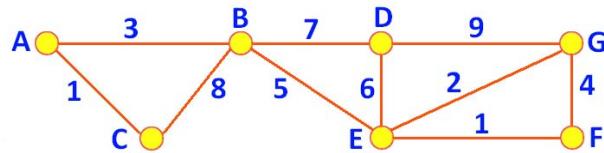


File Format Results Utilities Window Help

Solution for Minimal Spanning Tree Problem SOLLIN-ARTICULOS

	From Node	Connect To	Distance/Cost		From Node	Connect To	Distance/Cost
1	A	B	10	6	J	G	8
2	G	C	7	7	C	H	8
3	H	D	6	8	I	F	5
4	E	F	9	9	J	I	6
5	F	B	8				
	Total	Minimal	Connected	Distance or Cost	=		67

- SOLLIN: Encontrar un árbol generador mínimo del grafo ponderado.



Se elige inicialmente el elemento de menor valor. En este caso hay tres con valor 1, se selecciona uno de ellos, por ejemplo el 1 de la columna F y se marca, se elimina la columna donde se encuentra y se marca la fila F como la primera.

	A	B	C	D	E	F	G
	A		3	1			
	B	3		8	7	5	
	C	1	8				
	D		7		6		9
	E		5		6	1	2
<u>1</u> →	F				1		4
	G			9	2	4	

En la fila F se marca el 1, se elimina la columna E y se marca la fila E como segunda.

	A	B	C	D	E	F	G
	A		3	1			
	B	3		8	7	5	
	C	1	8				
	D		7		6		9
<u>2</u> →	E		5		6	1	2
<u>1</u> →	F				1		4
	G			9	2	4	

De filas marcadas (E, F) se elige el elemento con menor valor, no marcado anteriormente. En la fila E está el elemento 2, se marca y se elimina la columna G donde se encuentra, quedando la fila G como tercera.

	A	B	C	D	E	F	G
	A		3	1			
	B	3		8	7	5	
	C	1	8				
	D		7			6	9
<u>2</u> →	E		5		6		
<u>1</u> →	F					1	4
<u>3</u> →	G				9	2	4

De filas marcadas (E, F, G) se elige el elemento con menor valor, no marcado anteriormente. En la fila E está el elemento 5, se marca y se elimina la columna B donde se encuentra, quedando la fila B como cuarta.

	A	B	C	D	E	F	G
	A		3	1			
	B	3		8	7	5	
	C	1	8				
	D		7			6	9
<u>2</u> →	E		5		6		
<u>1</u> →	F					1	4
<u>3</u> →	G				9	2	4

De filas marcadas (B, E, F, G) se elige el elemento con menor valor, no marcado anteriormente. En la fila B está el elemento 3, se marca y se elimina la columna A donde se encuentra, quedando la fila A como quinta.

	A	B	C	D	E	F	G
	A		3	1			
	B	3		8	7	5	
	C	1	8				
	D		7			6	9
<u>2</u> →	E		5		6		
<u>1</u> →	F					1	4
<u>3</u> →	G				9	2	4

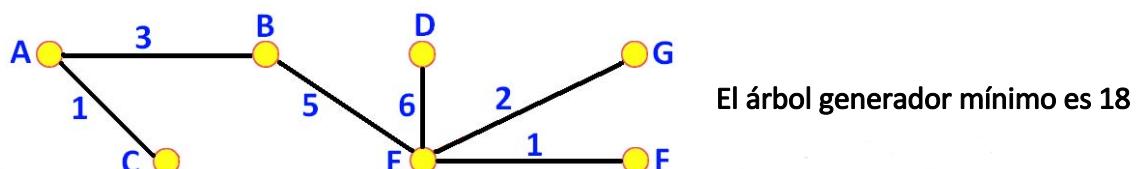
De filas marcadas (A, B, E, F, G) se elige el elemento con menor valor, no marcado anteriormente. En la fila A está el elemento 1, se marca y se elimina la columna C donde se encuentra, quedando la fila C como sexta.

	A	B	C	D	E	F	G
5 →	A		3	1			
4 →	B	3		8	7	5	
6 →	C	1	8				
	D		7		6		9
2 →	E		5		6	1	2
1 →	F				1		4
3 →	G			9	2	4	

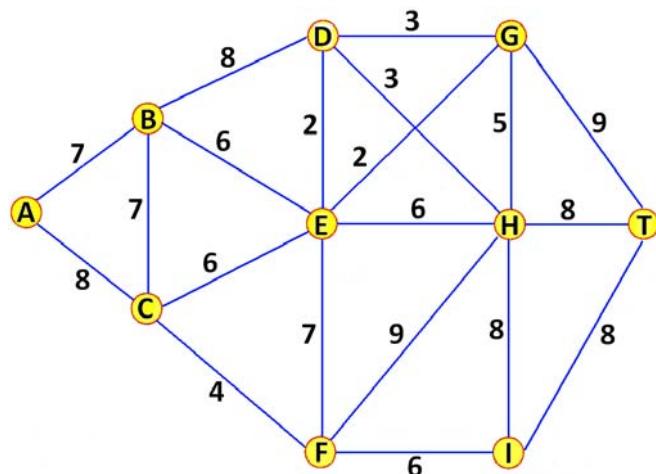
Finalmente, el elemento con menor valor de la columna D es el 6 (fila E) se marca y se elimina la columna D, queda la fila D como séptima.

	A	B	C	D	E	F	G
5 →	A		3	1			
4 →	B	3		8	7	5	
6 →	C	1	8				
7 →	D		7		6		9
2 →	E		5		6	1	2
1 →	F				1		4
3 →	G			9	2	4	

En un diagrama:



 **SOLLIN:** El grafo adjunto muestra una ruta aérea donde la distancia entre cada par de ciudades se expresa en cientos de millas. Obtener una red que conecte todas las ciudades y sea de mínima distancia.

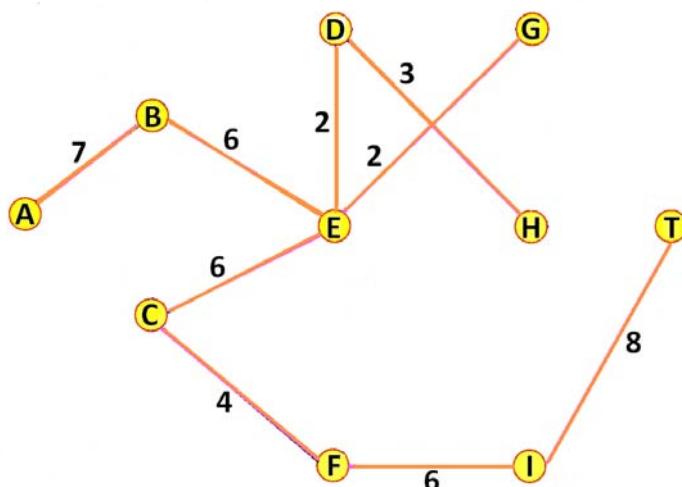


Al iniciar el algoritmo se elige el elemento de menor valor. En este caso hay tres elementos con valor 2, se selecciona arbitrariamente uno de ellos, por ejemplo el 2 de la columna D y se marca. Se elimina la columna D y se marca la fila D como la primera.

	A	B	C	D	E	F	G	H	I	T
9 →	A		7	8						
7 →	B	7 ⁹		7	8	6				
5 →	C	8	7			6	4 ⁶			
1 →	D		8			2 ²		3	3 ⁴	
2 →	E		6 ⁷	6 ⁵	2 ¹		7	2 ³	6	
6 →	F			4		7			9	6 ⁸
3 →	G				3	2			5	
4 →	H				3	6	9	5		8
8 →	I						6		8	
10 →	T							9	8	8 ¹⁰

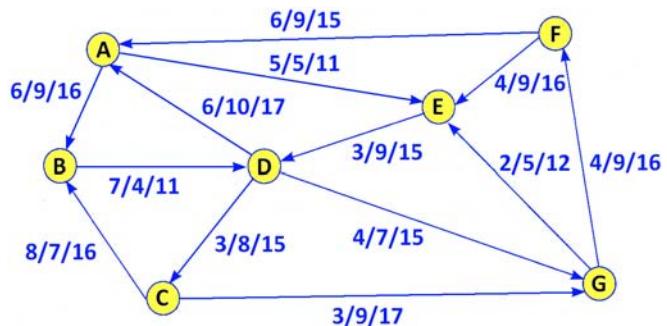
Una ruta aérea que conecte todas las ciudades y minimice la distancia tiene un valor de 44 millas.

Otra posibilidad sería intercambiar la ruta IT por HT.



H SOLLIN: Siete estaciones ferroviarias se conectan a través de una red provincial para intercambiar información. Por diversas circunstancias, las líneas no son iguales en cuanto a velocidad ni a coste de transmisión, variando la distancia de unos puestos a otros.

En el grafo adjunto se muestra la distribución actual de las líneas de comunicaciones, donde los nodos reflejan cada uno de los puestos de cada estación y las etiquetas de cada arista representan tres factores (longitud de las líneas, velocidad de transmisión y coste de envío de información por línea).

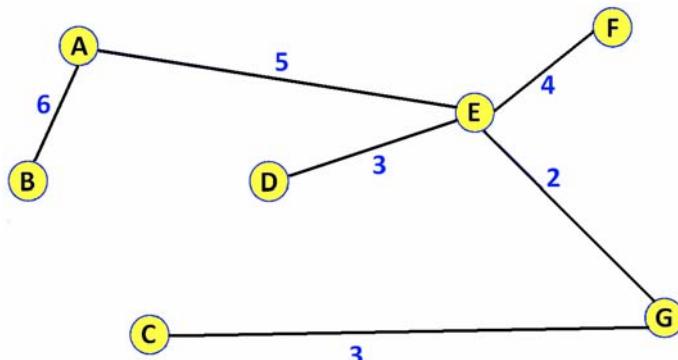


Para obtener una optimización de la línea, suponiendo que las líneas fueran bidireccionales y no fuera deseable que se formasen ciclos, la empresa ferroviaria contrata a una consultoría de comunicaciones para que dé respuestas a las preguntas siguientes:

- ¿Qué tendido mínimo de cable es necesario para unir todas las estaciones?
- ¿Con qué configuración se lograría conectar todas las estaciones a la mayor velocidad posible?

- a) Para obtener el mínimo tendido de cable (mínima longitud) se elabora la siguiente matriz de Sollin:

	A	B	C	D	E	F	G
6 →	A	6		6	5	6	
7 →	B	6	8	7			
3 →	C		8	³ 4			3
4 →	D	⁶ 6	⁷ 7	3			4
1 →	E	5			3		⁴ 5
5 →	F	6				4	4
2 →	G			³ 3	4	2 ¹	4



Una red de mínima longitud sería de 23

- b) Para configurar todas las estaciones a la mayor velocidad posible hay que encontrar el "árbol generador de máxima velocidad de transmisión", esto es, que la suma de los pesos de sus aristas sea máxima.

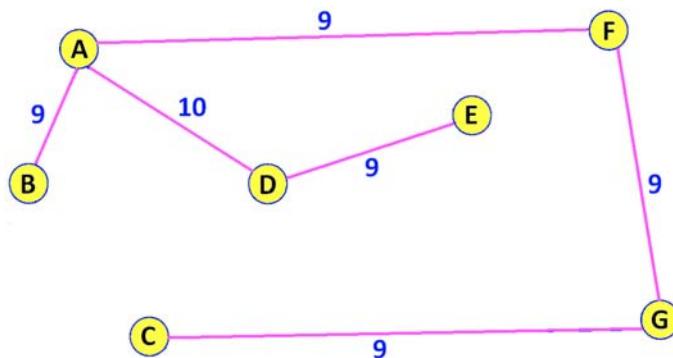
Para construir el árbol generador máximo puede emplearse de algoritmo de los pesos decrecientes como si se tratase de un árbol generador mínimo, con dos diferencias sustanciales:

- ☞ Añadir aristas en lugar de suprimirlas
- ☞ La conexión de una arista no debe formar un ciclo.

Es decir, el árbol generador máximo tendrá tantos vértices como el grafo.

Se construye la matriz de Sollin:

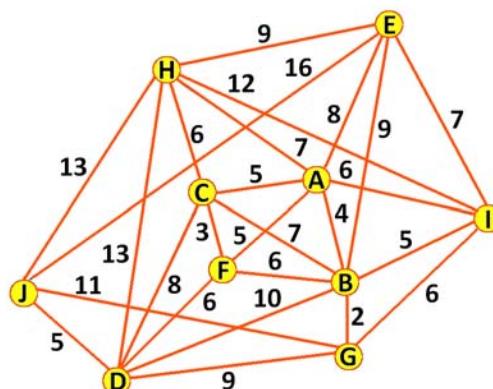
	A	B	C	D	E	F	G	
1 →	A		9^5		10^2	5	9^4	
5 →	B	9		7	4			
7 →	C		7		8		9^6	
2 →	D	10^1	4	8		9^3		7
3 →	E	10			9		9	5
4 →	F	9				9		9
6 →	G			9^7	7	5	9	



Una red de máxima velocidad es de 55

SOLLIN: El grafo adjunto presenta el diseño de un metro para Valladolid, indicando las estaciones (nodos) y las distancias que las separan en centenares de metros.

Suponiendo que el coste de la construcción es proporcional a la distancia y que cuando dos estaciones no se encuentran unidas es debido a imposibilidades técnicas por el río Pisuerga o a un coste excesivo.



- Diseñar la red de metro de forma que las estaciones queden conectadas a coste mínimo. ¿Cuál será la red a construir y cuál su longitud?
- El Ayuntamiento de Valladolid impone la condición de que entre dos estaciones de metro no tiene que haber más de cuatro estaciones intermedias, ¿Cuál sería la red de metro?
- Se trata de construir un árbol generador mínimo para la red conexa de la figura. Para ello, se aplica el algoritmo de Sollin, comenzando por establecer la matriz de incidencia asociada al grafo.

	A	B	C	D	E	F	G	H	I	J
<u>3</u> →	A		4	5 ⁴		8	5		7	6
<u>2</u> →	B	4 ³		7	10	9	6	2 ¹		5
<u>4</u> →	C	5	7		8		3		6	
	D		10	8			6	9	13	
	E	8	9						9	7
	F	5	6	3	6					
<u>1</u> →	G		2 ²		9				6	11
	H	7		6	13	9			12	13
	I	6	5			7		6	12	
	J				5	16		11	13	

3^a 2^a 4^a 1^a

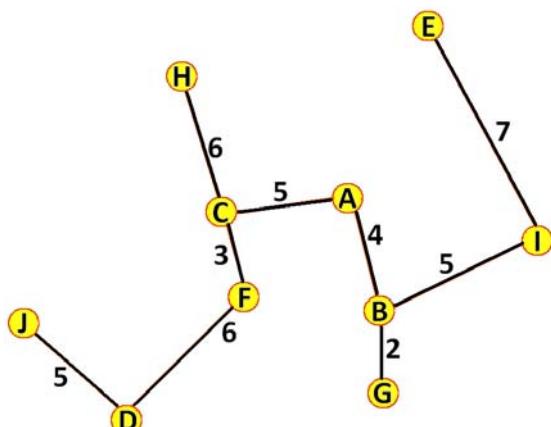
	A	B	C	D	E	F	G	H	I	J
3 →	A		4	5 ⁴		8	5		7	6
2 →	B	4 ³		7	10	9	6	2 ¹		5 ⁶
4 →	C	5	7		8		3 ⁵		6	
	D		10	8			6	9	13	5
	E	8	9						9	7
5 →	F	5	6	3	6					
1 →	G		2 ²		9				6	11
	H	7		6	13	9			12	13
6 →	I	6	5			7		6	12	
	J				5	16		11	13	
		3 ^a	2 ^a	4 ^a		5 ^a	1 ^a		6 ^a	

	A	B	C	D	E	F	G	H	I	J
3 →	A		4	5 ⁴		8	5		7	6
2 →	B	4 ³		7	10	9	6	2 ¹		5 ⁶
4 →	C	5	7		8		3 ⁵		6 ⁷	
8 →	D		10	8			6	9	13	5 ⁹
10 →	E	8	9						9	7
5 →	F	5	6	3	6 ⁸					
1 →	G		2 ²		9				6	11
7 →	H	7		6	13	9			12	13
6 →	I	6	5			7 ¹⁰		6	12	
9 →	J				5	16		11	13	
		3 ^a	2 ^a	4 ^a	8 ^a	10 ^a	5 ^a	1 ^a	7 ^a	6 ^a
										9 ^a

UNICAMENTE PARA USO ESCOLAR
(IMPRESO EN PAPEL RECICLADO)

El árbol generador mínimo indica la red de metro a construir, con una longitud de $43 \times 100 = 4300$ metros

Analizando el árbol generador mínimo, la cadena JDFCABIE no cumple la condición, la estación E se encuentra separado de la estación J por más de cuatro estaciones.



b) La red de metro diseñada no cumple las condiciones del Ayuntamiento de Valladolid de que entre dos estaciones de metro no tiene que haber más de cuatro estaciones intermedias.

En la red de metro construida, la arista de mayor valor es IE con un peso de 700 metros.

Una vez eliminada la arista IE, al intentar substituirla, no se encuentra otra estación que siga conservando la estructura del árbol.

En caso de no ser posible, se elige otra arista de un valor inmediatamente superior que no pertenezca al árbol. Se elige la arista AE con peso 800 metros.

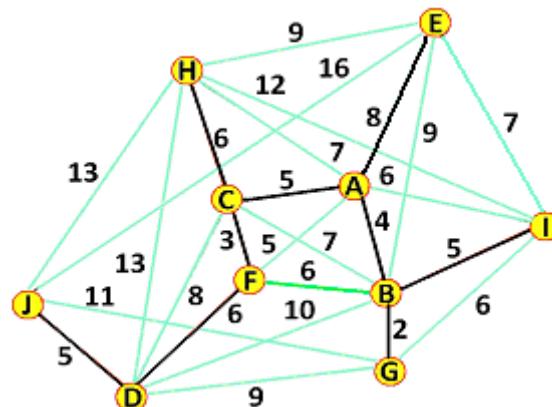
La cadena JDFCAEBI tampoco cumple la condición, la arista de mayor valor es DF.

Al eliminarla y querer susbsituirla no se encuentra otra arista de ese valor que conserve la estructura del árbol.

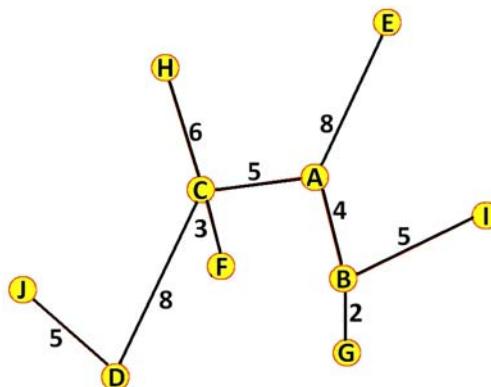
Se elige otra arista de un valor inmediatamente superior que no pertenezca al árbol.

Se selecciona la arista DC.

En cualquier caso, la arista elegida no se añade directamente al árbol hasta verificar que proporciona una posible cadena. Una vez que ha sido obtenida, la arista no se elimina del árbol en posteriores iteraciones del proceso



Con las exigencias del Ayuntamiento, la red de metro será la de la figura adjunta, con una longitud de $46 \times 100 = 4600$ metros

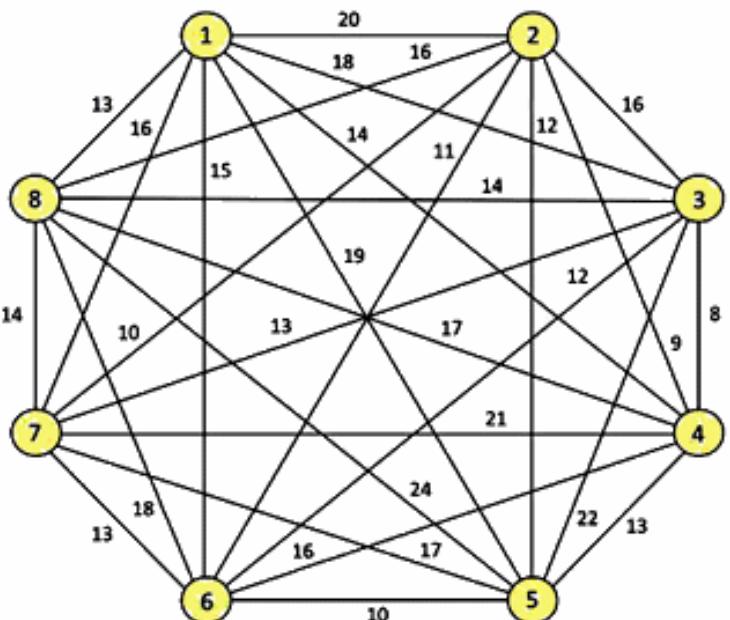


Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

SOLLIN (ÁRBOL EXPANSIÓN MÍNIMA)

Una empresa aeronáutica decide instalar un sistema de distribución que permita enviar contenedores a ocho provincias españolas, considerando la posibilidad de envío no directo. Los costes en cientos de euros se reflejan en la tabla adjunta. Calcular el coste mínimo.

	1	2	3	4	5	6	7	8
1	—	20	18	14	19	15	16	13
2	20	—	16	9	12	11	10	16
3	18	16	—	8	22	12	13	14
4	14	9	8	—	13	16	21	17
5	19	12	22	13	—	10	17	24
6	15	11	12	16	10	—	13	18
7	16	10	13	21	17	13	—	14
8	13	16	14	17	24	18	14	—



GRAFO ADJUNTO:

Se selecciona el mínimo elemento de la tabla (elemento 8) que se encuentra en la casilla (3, 4). En caso de existir varios, se selecciona uno cualquiera.

Se eliminan las columnas 3 y 4, marcando las filas 3 y 4.

1	2	3	4	5	6	7	8
1	—	20	18	14	19	15	16
2	20	—	16	9	12	11	10
3	18	16	—	8	22	12	13
4	14	9	8	—	13	16	21
5	19	12	22	13	—	10	17
6	15	11	12	16	10	—	13
7	16	10	13	21	17	13	—
8	13	16	14	17	24	18	—

1	2	3	4	5	6	7	8
1	—	20		19	15	16	13
2	20	—		12	11	10	16
3	18	16		22	12	13	14
4	14	9		13	16	21	17
5	19	12		—	10	17	24
6	15	11		10	—	13	18
7	16	10		17	13	—	14
8	13	16		24	18	14	—

Como no están marcadas todas las filas, el algoritmo sigue. Se selecciona el mínimo elemento que pertenece a las filas marcadas y no pertenecen a las columnas eliminadas.

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

El mínimo elemento es el 9, que se encuentra en la casilla (4, 2), se elimina la columna 2 y se marca la fila 2.

	1	2	3	4	5	6	7	8		1	2	3	4	5	6	7	8
1	—	20			19	15	16	13	1	—				19	15	16	13
2	20	—			12	11	10	16	3•2	20				12	11	10	16
1•3	18	16			22	12	13	14	1•3	18				22	12	13	14
2•4	14	9			13	16	21	17	2•4	14				13	16	21	17
5	19	12			—	10	17	24	5	19				—	10	17	24
6	15	11			10	—	13	18	6	15				10	—	13	18
7	16	10			17	13	—	14	7	16				17	13	—	14
8	13	16			24	18	14	—	8	13				24	18	14	—

Se selecciona el mínimo elemento que pertenece a las filas marcadas y no pertenecen a las columnas eliminadas.

El mínimo elemento es el 10, que se encuentra en la casilla (2, 7), se elimina la columna 7 y se marca la fila 7.

	1	2	3	4	5	6	7	8		1	2	3	4	5	6	7	8
—					19	15	16	13	1	—				19	15	—	13
3•	20				12	11	10	16	3•2	20				12	11	—	16
1•	18				22	12	13	14	1•3	18				22	12	—	14
2•	14				13	16	21	17	2•4	14				13	16	—	17
19					—	10	17	24	5	19				—	10	—	24
15					10	—	13	18	6	15				10	—	—	18
16					17	13	—	14	4•7	16				17	13	—	14
13					24	18	14	—	8	13				24	18	—	—

Se selecciona el mínimo elemento que pertenece a las filas marcadas y no pertenecen a las columnas eliminadas.

El mínimo elemento es el 11, que se encuentra en la casilla (2, 6), se elimina la columna 6 y se marca la fila 6.

	1	2	3	4	5	6	7	8		1	2	3	4	5	6	7	8
1	—				19	15		13	1	—				19		—	13
3•2	20				12	11	10	16	3•2	20				12	—	—	16
1•3	18				22	12		14	1•3	18				22	—	—	14
2•4	14				13	16		17	2•4	14				13	—	—	17
5	19				—	10		24	5	19				—	—	—	24
6	15				10	—		18	5•6	15				10	—	—	18
4•7	16				17	13		14	4•7	16				17	—	—	14
8	13				24	18		—	8	13				24	—	—	—

Se selecciona el mínimo elemento que pertenece a las filas marcadas y no pertenecen a las columnas eliminadas.

El mínimo elemento es el 10, que se encuentra en la casilla (6, 5), se elimina la columna 5 y se marca la fila 6.

	1	2	3	4	5	6	7	8		1	2	3	4	5	6	7	8
1	—			19		13			1	—							13
3• 2	20			12		16	3• 2	20									16
1• 3	18			22		14	1• 3	18									14
2• 4	14			13		17	2• 4	14									17
5	19			—		24	6• 5	19									24
5• 6	15			10		18	5• 6	15									18
4• 7	16			17		14	4• 7	16									14
8	13			24		—	8	13									—

Se selecciona el mínimo elemento que pertenece a las filas marcadas y no pertenecen a las columnas eliminadas.

El mínimo elemento es el 14, que se encuentra en la casilla (7, 8), se elimina la columna 8 y se marca la fila 8.

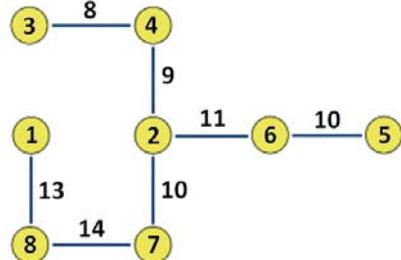
	1	2	3	4	5	6	7	8		1	2	3	4	5	6	7	8
1	—				13				1	—							
3• 2	20				16	3• 2	20										
1• 3	18				14	1• 3	18										
2• 4	14				17	2• 4	14										
6• 5	19				24	6• 5	19										
5• 6	15				18	5• 6	15										
4• 7	16				14	4• 7	16										
8	13				—	7• 8	13										

El mínimo elemento es el 13, que se encuentra en la casilla (8, 1), se elimina la columna 1 y se marca la fila 1.

	1	2	3	4	5	6	7	8		1	2	3	4	5	6	7	8
1	—				8• 1				1	—							
3• 2	20				3• 2												
1• 3	18				1• 3												
2• 4	14				2• 4												
6• 5	19				6• 5												
5• 6	15				5• 6												
4• 7	16				4• 7												
7• 8	13				7• 8												

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

	1	2	3	4	5	6	7	8
8° 1	-	20	18	14	19	15	16	13
3° 2	20	-	16	9	12	11	10	16
1° 3	18	16	-	8	22	12	13	14
2° 4	14	9	8	-	13	16	21	17
6° 5	19	12	22	13	-	10	17	24
5° 6	15	11	12	16	10	-	13	18
4° 7	16	10	13	21	17	13	-	14
7° 8	13	16	14	17	24	18	14	-



Longitud o peso mínimo de la distribución: $8 + 9 + 11 + 10 + 10 + 14 + 13 = 75$

Coste mínimo de la distribución: $75 \times 100 = 7.500$ euros

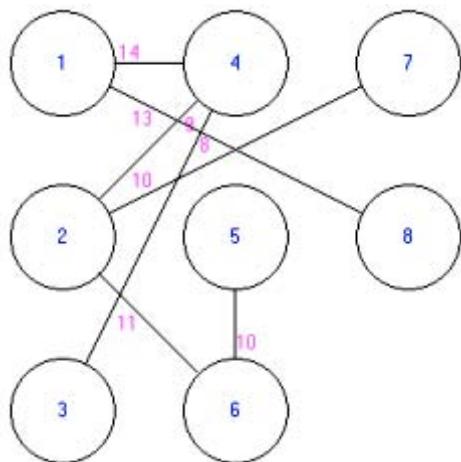
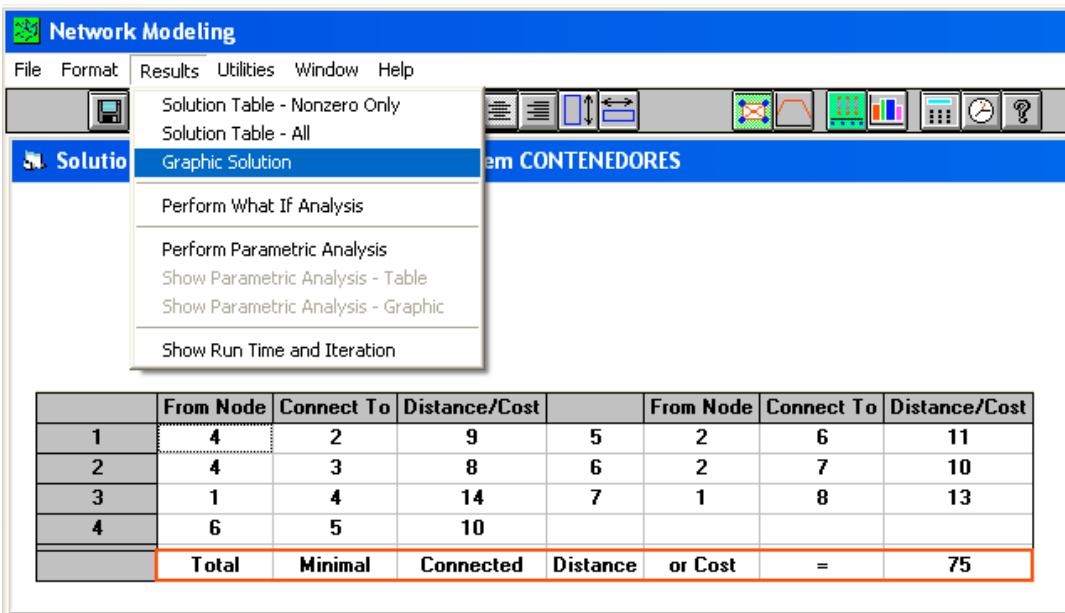
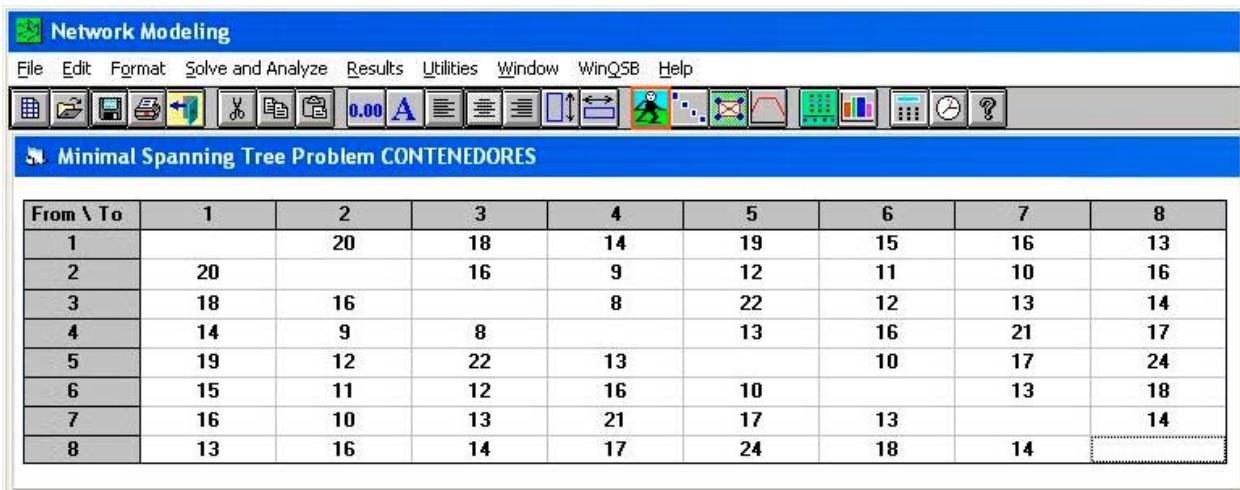


WinQSB / Network Modeling - Minimal Spanning Tree

NET Problem Specification

Problem Type	Objective Criterion	
<input type="radio"/> Network Flow <input type="radio"/> Transportation Problem <input type="radio"/> Assignment Problem <input type="radio"/> Shortest Path Problem <input type="radio"/> Maximal Flow Problem <input checked="" type="radio"/> Minimal Spanning Tree <input type="radio"/> Traveling Salesman Problem	<input checked="" type="radio"/> Minimization <input type="radio"/> Maximization	
Data Entry Format		
<input checked="" type="radio"/> Spreadsheet Matrix Form <input type="radio"/> Graphic Model Form <input checked="" type="checkbox"/> Symmetric Arc Coefficients <small>[i.e., both ways same cost]</small>		
Problem Title	CONTENEDORES	
Number of Nodes	8	
OK	Cancel	Help

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

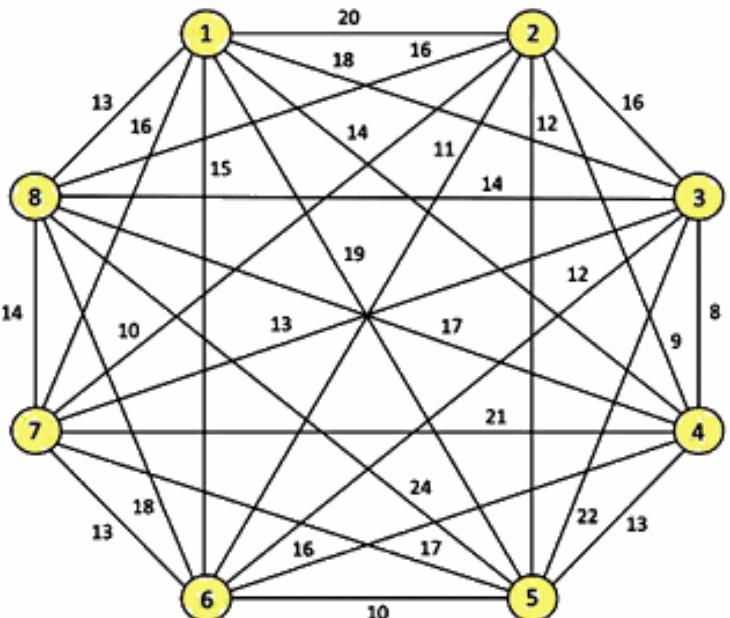


Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

SOLLIN (ÁRBOL EXPANSIÓN MÁXIMA)

Una empresa aeronáutica decide instalar un sistema de distribución que permite enviar contenedores a ocho provincias españolas, considerando la posibilidad de envío no directo. Los costes en cientos de euros se reflejan en la tabla adjunta. Calcular el máximo coste.

	1	2	3	4	5	6	7	8
1	—	20	18	14	19	15	16	13
2	20	—	16	9	12	11	10	16
3	18	16	—	8	22	12	13	14
4	14	9	8	—	13	16	21	17
5	19	12	22	13	—	10	17	24
6	15	11	12	16	10	—	13	18
7	16	10	13	21	17	13	—	14
8	13	16	14	17	24	18	14	—



GRAFO ADJUNTO:

Se selecciona el máximo elemento de la tabla (elemento 24) que se encuentra en la casilla (8, 5).

Se eliminan las columnas 5 y 8, y se marcan las filas 5 y 8.

En caso de existir varios, se selecciona uno cualquiera.

	1	2	3	4	5	6	7	8
1	—	20	18	14	19	15	16	13
2	20	—	16	9	12	11	10	16
3	18	16	—	8	22	12	13	14
4	14	9	8	—	13	16	21	17
5	19	12	22	13	—	10	17	24
6	15	11	12	16	10	—	13	18
7	16	10	13	21	17	13	—	14
8	13	16	14	17	24	18	14	—

	1	2	3	4	5	6	7	8
1	—	20	18	14	—	15	16	
2	20	—	16	9	—	11	10	
3	18	16	—	8	—	12	13	
4	14	9	8	—	—	16	21	
5	19	12	22	13	—	—	10	17
6	15	11	12	16	—	—	—	13
7	16	10	13	21	17	—	13	—
8	13	16	14	17	24	18	14	—

Como no están marcadas todas las filas, el algoritmo continúa.

Se selecciona el máximo elemento que pertenece a las filas marcadas y no pertenecen a las columnas eliminadas.

El máximo elemento es el 22, que se encuentra en la casilla (5, 3), se elimina la columna 3 y se marca la fila 3.

	1	2	3	4	5	6	7	8		1	2	3	4	5	6	7	8	
1	—	20	18	14		15	16		1	—	20		14		15	16		
2	20	—	16	9		11	10		2	20	—		9		11	10		
3	18	16	—	8		12	13		3 ^a	3	18	16	—	8		12	13	
4	14	9	8	—		16	21		4	14	9		—		16	21		
1 ^a	5	19	12	22	13	—	10	17	1 ^a	5	19	12		13	—	10	17	
6	15	11	12	16		—	13		6	15	11		16		—	13		
7	16	10	13	21		13	—		7	16	10		21		13	—		
2 ^a	8	13	16	14	17		18	14	—	2 ^a	8	13	16		17	18	14	—

Se selecciona el máximo elemento que pertenece a las filas marcadas y no pertenecen a las columnas eliminadas.

El máximo elemento es el 19, que se encuentra en la casilla (5, 1), se elimina la columna 1 y se marca la fila 1.

	1	2	3	4	5	6	7	8		1	2	3	4	5	6	7	8	
1	—	20		14		15	16		4 ^a	1	—	20		14		15	16	
2	20	—		9		11	10		2	—		9		11	10			
3 ^a	3	18	16	—	8	12	13		3 ^a	3	16	—	8		12	13		
4	14	9		—		16	21		4	9		—		16	21			
1 ^a	5	19	12		13	—	10	17	1 ^a	5	12		13	—	10	17		
6	15	11		16		—	13		6	11		16		—	13			
7	16	10		21		13	—		7	10		21		13	—			
2 ^a	8	13	16		17	18	14	—	2 ^a	8	16		17		18	14	—	

Se selecciona el máximo elemento que pertenece a las filas marcadas y no pertenecen a las columnas eliminadas.

El máximo elemento es el 20, que se encuentra en la casilla (1, 2), se elimina la columna 2 y se marca la fila 2.

	1	2	3	4	5	6	7	8		1	2	3	4	5	6	7	8	
4 ^a	1	—	20		14		15	16	4 ^a	1	—		14		15	16		
2	—		9		11	10			5 ^a	2	—		9		11	10		
3 ^a	3	16	—	8		12	13		3 ^a	3	—		8		12	13		
4	9		—		16	21			4		—		16		21			
1 ^a	5	12		13	—	10	17		1 ^a	5			13	—	10	17		
6	11		16		—	13			6			16		—	13			
7	10		21		13	—			7			21		13	—			
2 ^a	8	16		17	18	14	—		2 ^a	8		17		18	14	—		

Se selecciona el máximo elemento que pertenece a las filas marcadas y no pertenecen a las columnas eliminadas.

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

El máximo elemento es el 18, que se encuentra en la casilla (8, 6), se elimina la columna 6 y se marca la fila 6.

	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
4 ^a 1	—			14		15	16		4 ^a 1	—		14		16		
5 ^a 2		—		9		11	10		5 ^a 2		—	9		10		
3 ^a 3			—	8		12	13		3 ^a 3			8		13		
4				—		16	21		4			—		21		
1 ^a 5				13	—	10	17		1 ^a 5			13	—	17		
6				16		—	13		6 ^a 6			16		13		
7				21		13	—		7			21		—		
2 ^a 8				17		18	14	—	2 ^a 8			17		14		

Se selecciona el máximo elemento que pertenece a las filas marcadas y no pertenecen a las columnas eliminadas.

El máximo elemento es el 17, que se encuentra en la casilla (8, 4), se elimina la columna 4 y se marca la fila 4.

	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
4 ^a 1	—			14		16			4 ^a 1	—		16				
5 ^a 2		—		9		10			5 ^a 2		—	10				
3 ^a 3			—	8		13			3 ^a 3			13				
4				—		21			7 ^a 4			21				
1 ^a 5				13	—	17			1 ^a 5			17				
6 ^a 6				16		13			6 ^a 6			13				
7				21		—			7			—				
2 ^a 8				17		14	—		2 ^a 8			14	—			

Se selecciona el máximo elemento que pertenece a las filas marcadas y no pertenecen a las columnas eliminadas.

El máximo elemento es el 21, que se encuentra en la casilla (4, 7), se elimina la columna 7 y se marca la fila 7.

Asignatura Grupo
 Apellidos Nombre
 Ejercicio del día

	1	2	3	4	5	6	7	8		1	2	3	4	5	6	7	8
4 ^a 1	-						16		4 ^a 1	-							
5 ^a 2	-						10		5 ^a 2	-							
3 ^a 3	-						13		3 ^a 3	-							
7 ^a 4		-					21		7 ^a 4		-						
1 ^a 5		-					17		1 ^a 5		-						
6 ^a 6							13		6 ^a 6								
7							-		8 ^a 7					-			
2 ^a 8							14	-	2 ^a 8					-			

El algoritmo ha finalizado al marcar todas las filas.

	1	2	3	4	5	6	7	8	
Longitud máxima del árbol:	1	-	20	18	14	19	15	16	13
19 + 20 + 22 + 17 + 18 + 21 + 24 = 141	2	20	-	16	9	12	11	10	16
El coste máximo de la distribución:	3	18	16	-	8	22	12	13	14
141 x 100 = 14.100 euros.	4	14	9	8	-	13	16	21	17
	5	19	12	22	13	-	10	17	24
	6	15	11	12	16	10	-	13	18
	7	16	10	13	21	17	13	-	14
	8	13	16	14	17	24	18	14	-

Asignatura **Grupo**
Apellidos **Nombre**
Ejercicio del día

Asignatura Grupo
Apellidos Nombre
Ejercicio del día



Instrumentos Estadísticos Avanzados
Facultad Ciencias Económicas y Empresariales
Departamento de Economía Aplicada
Profesor: Santiago de la Fuente Fernández