0. Ambientación

En general, un grafo es una manera de representar relaciones que existen entre pares de objetos. Así, un grafo es un conjunto de objetos, llamados vértices[1], y relaciones entre objetos que establecen una relación entre pares de vértices, representadas por aristas.

Definición 1. Un grafo se define como un par G = (V, A), donde V es un conjunto finito no vacío de vértices y A es un conjunto de pares de vértices de V, es decir, las aristas.

Definición 2. Un grafo G se define como un par ordenado, G = (V, A), donde V es un conjunto finito y A es un conjunto que consta de dos elementos de V.

[1] La terminología de la teoría de grafos NO es estándar. El concepto de vértice también se referencia como nodo. Asimismo, aristas (edges en inglés) y arcos denotan el mismo elemento.

En algunos libros, sin embargo, se establece una diferencia entre aristas (unen vértices en un grafo no dirigido) y arcos (unen vértices en grafos dirigidos). En este documento, se dará preferencia a los términos vértice y arista.

PARTE 1. CONCEPTOS FUNDAMENTALES

1. Grafos dirigidos y no dirigidos. Dependiendo del tipo de relación entre los vértices del grafo, se definen distintos tipos de grafos. Así se distinguen aristas dirigidas y no dirigidas:

Arista dirigida: es aquella que define un par ordenado de vértices (u , v), donde el primer vértice u es el origen de la arista y el segundo vértice v es el término (o vértice final).

El par (u , v) ≠ (v , u).

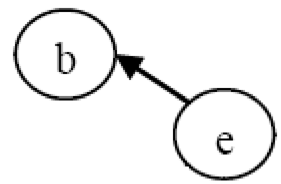


Figura 1. Ejemplo de arista dirigida.

Arista no dirigida: aquella que define un par no ordenado de vértices (u , v), donde (u , v) = (v , u).

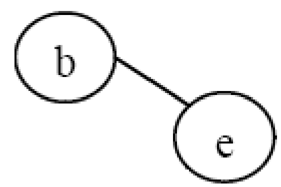


Figura 2. Ejemplo de arista NO dirigida.

De esta forma se distinguen entre grafos dirigidos y grafos no dirigidos.

Grafo dirigido: Es aquel cuyas aristas son dirigidas. Los grafos dirigidos suelen representar relaciones asimétricas, por ejemplo: relaciones de herencia, red de agua (acueducto), etc.

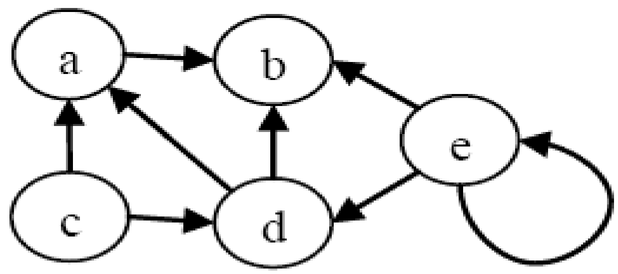


Figura 3. Ejemplo de Grafo Dirigido.

Grafo no dirigido: Es aquel cuyas aristas son no dirigidas. Representan relaciones simétricas como relaciones de hermandad y colaboración, conexiones de transportes, etc.

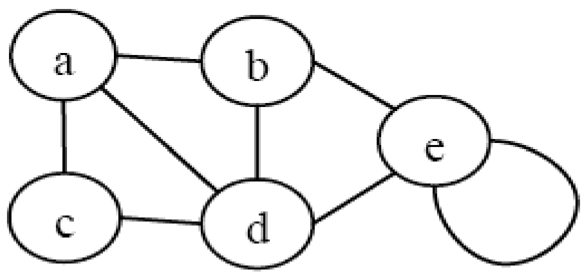


Figura 4. Ejemplo de Grafo NO dirigido.

2. Incidencia, adyacencia y grado de un vértice. Sea un grafo G = (V, A), los vértices u y v pertenecientes a V; y una arista (u,v) perteneciente a A, se tiene:

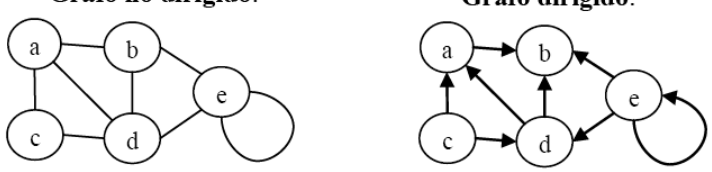
Incidencia: la arista (u,v) es incidente con los vértices u y v.

Adyacencia: Dos vértices u y v son adyacentes si existe una arista que permita:
El vértice u es adyacente a v **Y** el vértice v es adyacente desde u

Grado: El grado de un vértice u es el número de vértices adyacentes a u. Para un grafo dirigido, el grado de salida de un vértice u es el número de vértices adyacentes desde u, mientras que el grado de entrada de un vértice u es el número de vértices adyacentes a u.

La Figura 5 muestra los grados de los vértices para un grafo no dirigido y un grafo dirigido.

Grafo no dirigido: **Grafo dirigido**:



Grado (a) = 3
Grado (b) = 3
Grado (c) = 2
Grado (d) = 4
Grado (e) = 4

GradoE (a) = 2	GradoS (a) = 1
GradoE (b) = 3	GradoS (b) = 0
GradoE (c) = 0	GradoS (c) = 2
GradoE (d) = 2	GradoS (d) = 2
GradoE (e) = 1	GradoS (e) = 3

Figura 5. Grado de los vértices en un grafo no dirigido y dirigido.

La suma de los grados de los vértices es el doble del número de aristas.

A esta propiedad se le conoce como el "Teorema de los apretones de manos".

3. Grafos valorados y grafos etiquetados[2]. Un **grafo valorado** (o ponderado) es una TERNA <V, A, f> donde <V, A> es un grafo y f es una función cualquiera, denominada **función de coste**, que asocia un valor o peso a cada arista en el grafo. El peso de un camino en un grafo con pesos es la suma de los pesos de todas las aristas atravesadas.

En un **grafo etiquetado**, la función f tiene como imagen un conjunto de **etiquetas no numéricas**.

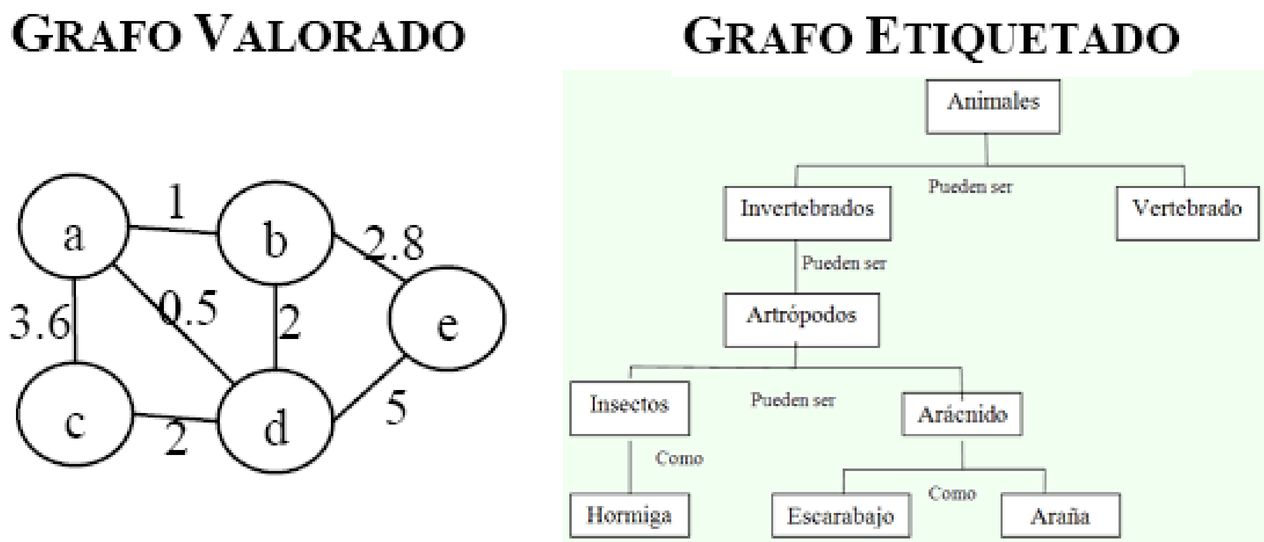


Figura 6. Ejemplo de Grafo Valorado y de Grafo Etiquetado.

[2] En este curso no se van a tratar los grafos valorados ni etiquetados.

4. Camino, Bucle y Ciclo. Es muy importante distinguir estos tres conceptos.

Un CAMINO es una secuencia que alterna vértices y aristas que comienza por un vértice y termina en vértice tal que cada arista es incidente a su vértice predecesor y sucesor. Es decir, un camino es una sucesión de vértices **V: <v_0, v_1, v_2, … v_k>** que cumple:

$$(v_i, v_{i+1}) \in A \ \forall \ i \in \{0 \ldots k-1\}$$

La ***longitud k*** del camino está dada por el número de aristas.

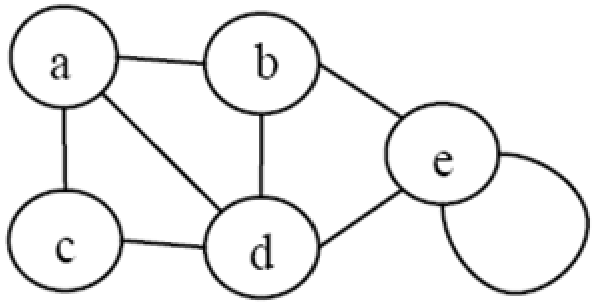
Un camino es simple **si cada vértice en el camino es distinto**, **excepto** posiblemente por el primero y el último vértice.

Un BUCLE es un camino de longitud 1 que comienza y termina en el mismo vértice.

Un CICLO o CIRCUITO es un camino simple **<v_0 … v_k>** que cumple las siguientes restricciones:

$v_0 = v_k$
Si es no dirigido, $k = 1$ (es un bucle) o $k \geq 3$.

GRAFO NO DIRIGIDO:



<a,c,d,a,b,e>: camino de longitud 5.

<a,b,e,d,c>: camino simple de longitud 4.

<b,e,d,b>: ciclo de longitud 3.

<a,e>: no es un camino.

<*e*,*e*>: camino, bucle y ciclo.

Figura 7. Caminos, ciclos y bucles en un grafo dirigido y no dirigido.

5. Grafo Conexo, Fuertemente y Débilmente. Se puede apreciar lo siguiente:

Un grafo no dirigido es conexo si existe un camino entre cada par de vértices.

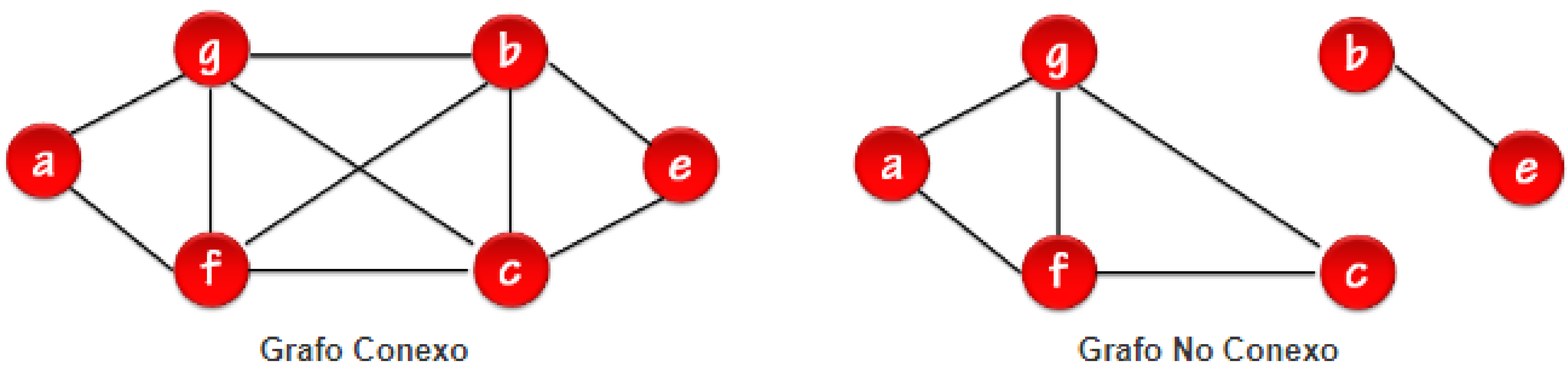


Figura 8. Ejemplo de Grafo No Dirigido Conexo y Grafo No Dirigido No Conexo.

6. Representaciones de un grafo. En esta sección se mencionan 3 estilos.

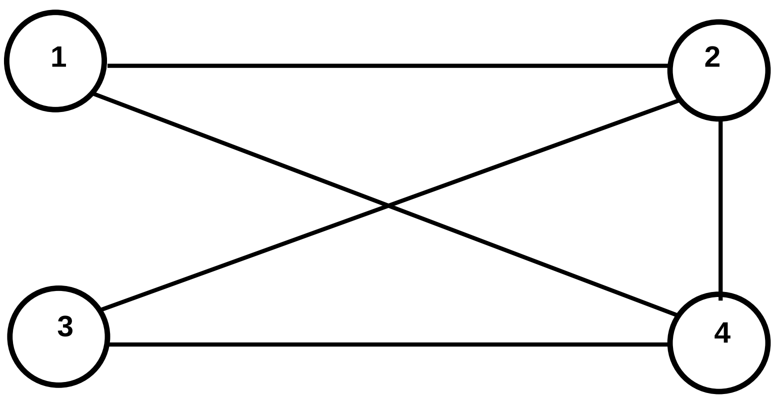
Matriz de adyacencia. La matriz de adyacencia de un grafo

$G = (V, E)$ es una matriz lógica **$A = (a_{ij})$** de orden VxV donde **$(a_{ij}) = 1$**

si existe una arista entre los vértices **V_i** y **V_j**.

En los grafos NO dirigidos normalmente la matriz A es simétrica[3].

En la matriz de adyacencia de un grafo NO dirigido la suma de los elementos de cada fila es igual al grado de ese vértice.



	1	**2**	**3**	**4**
1	0	**1**	0	**1**
2	**1**	0	**1**	**1**
3	0	**1**	0	**1**
4	**1**	**1**	**1**	0

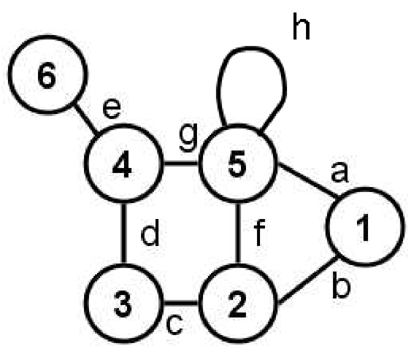
1: Existencia de la arista
0: NO Existencia de la arista

Figura 9. Ejemplo de un Grafo NO Dirigido y su matriz de adyacencia correspondiente.

[3] Una matriz es simétrica si es una matriz cuadrada, la cual tiene la característica de ser igual a su transpuesta.
Se llama matriz transpuesta de A a la matriz que se obtiene cambiando ordenadamente las filas por las columnas.

Matriz de incidencia. Sea G = {V,A} un grafo NO dirigido con **n** vértices y **m** aristas. Se denomina matriz de incidencia **I** a la matriz de orden **n** x **m** compuesta de los siguientes elementos:

0	si el vértice i	NO es incidente	con la arista j.
1	si el vértice i	es incidente	con la arista j.
2	si la arista j	es un bucle	en el vértice i.



	a	*b*	*c*	*d*	*e*	*f*	*g*	*h*
1	**1**	**1**	0	0	0	0	0	0
2	0	**1**	**1**	0	0	**1**	0	0
3	0	0	**1**	**1**	0	0	0	0
4	0	0	0	**1**	**1**	0	**1**	0
5	**1**	0	0	0	0	**1**	**1**	**2**
6	0	0	0	0	**1**	0	0	0

Como se puede apreciar, en las filas están los vértices del grafo, y en las columnas están las aristas.

Si se revisa, por ejemplo, el vértice 2 (fila 2), vemos que en la intersección con las columnas **b**, **c** y **f** tiene un 1, esto quiere decir, que el vértice 2 es incidente con las aristas **b**, **c** y **f**.

En el caso de los bucles se cuenta como 2 tal como sucede en la intersección de la fila 5 con la columna h.

Figura 10. Ejemplo de un Grafo NO Dirigido y su matriz de incidencia correspondiente.

Para las matrices de incidencia también se cumplen una serie de propiedades:

1. Si el grafo G es NO dirigido, la suma de los elementos de cada fila de la matriz de incidencia es igual al grado del correspondiente vértice.

2. Si el grafo G es NO dirigido, la suma de los elementos de cada columna de la matriz de incidencia debe ser igual a 2. Cada arista tiene un vértice en cada extremo, dos en total, no puede tener más.

Listas. La principal desventaja de usar la matriz de adyacencia es que para leer o examinar la matriz se requiere un tiempo de **$O(n^2)$**. Para evitar esta desventaja se puede usar otra representación llamada lista de adyacencia.

La lista de adyacencia de un vértice i es una lista, en algún orden, de todos los vértices adyacentes a i.

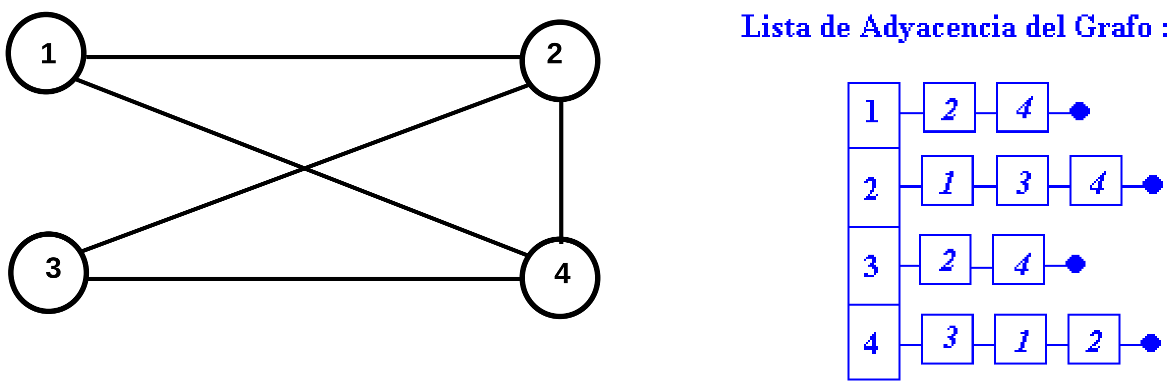


Figura 11. Ejemplo de un Grafo NO Dirigido y su Lista de adyacencia correspondiente.

La representación mediante lista de adyacencia se usa cuando el número de aristas es mucho menor a **n^2**.

Si el número de aristas es cercano a **n^2** entonces no hay mayor diferencia de desempeño si se compara a una lista de adyacencia con una matriz de adyacencia.

Desventaja: puede tomar un tiempo O(n) para determinar si hay una arista del vértice i al j.

7. Subgrafos y complementos. En esta sección se revisarán esos conceptos.

Subgrafo. La figura 12 permite observar dos grafos no dirigidos, G es un grafo que posee 4 vértices y 6 aristas, en este caso es el grafo original; mientras G' es el subgrafo de G. Como podemos observar G' cumple todas las condiciones de subgrafo, es decir, V' es subconjunto de V y E' es subconjunto de E.

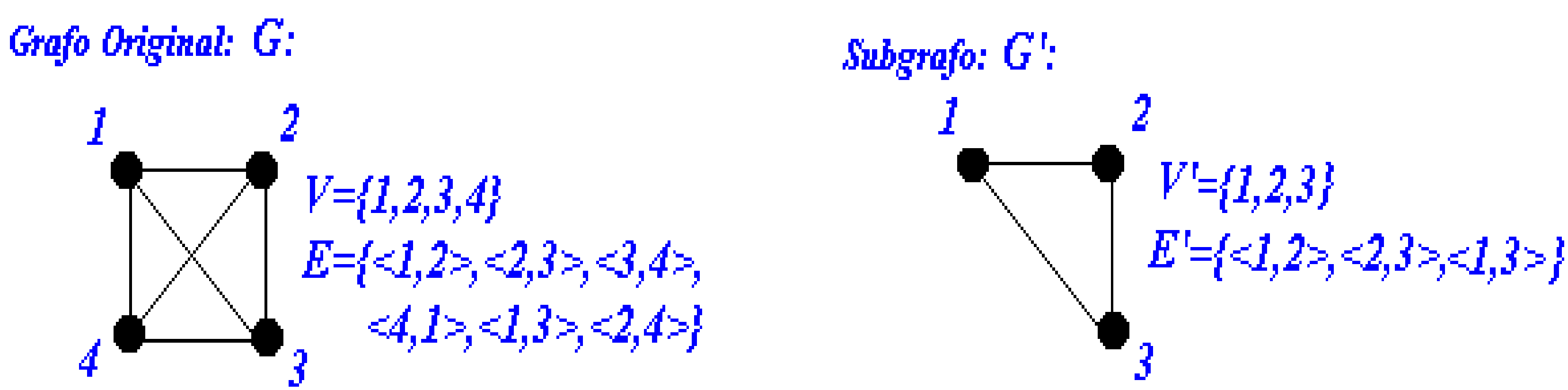


Figura 12. Ejemplo de un Grafo y un subgrafo

Subgrafo expandido. La figura 13 permite observar dos grafos no dirigidos, G es un grafo que posee 4 vértices y 6 aristas, en este caso es el grafo original; mientras G' es el subgrafo de G, pero en este caso es un subgrafo expandido. Como podemos observar G' cumple todas las condiciones de subgrafo, es decir, V' es subconjunto de V con la particularidad de que V' debe ser igual a V y E' es subconjunto de E.

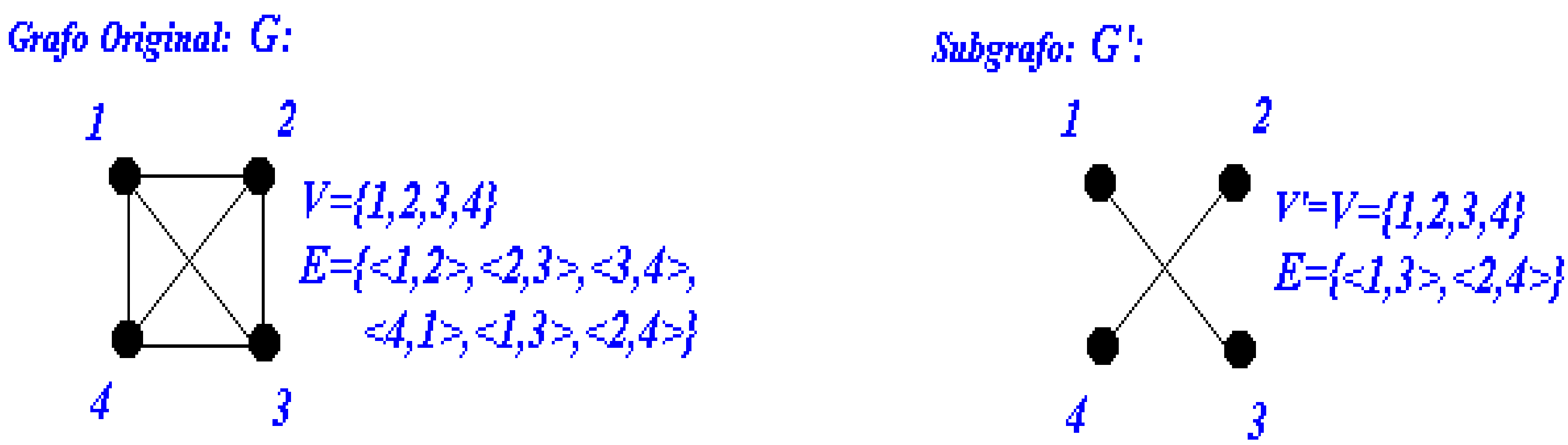


Figura 13. Ejemplo de un Grafo y un subgrafo expandido

<u>Grafo completo.</u> Un grafo completo tiene cada par de vértices unido por una arista. Se denota por **K_n** al grafo completo de **n** vértices.

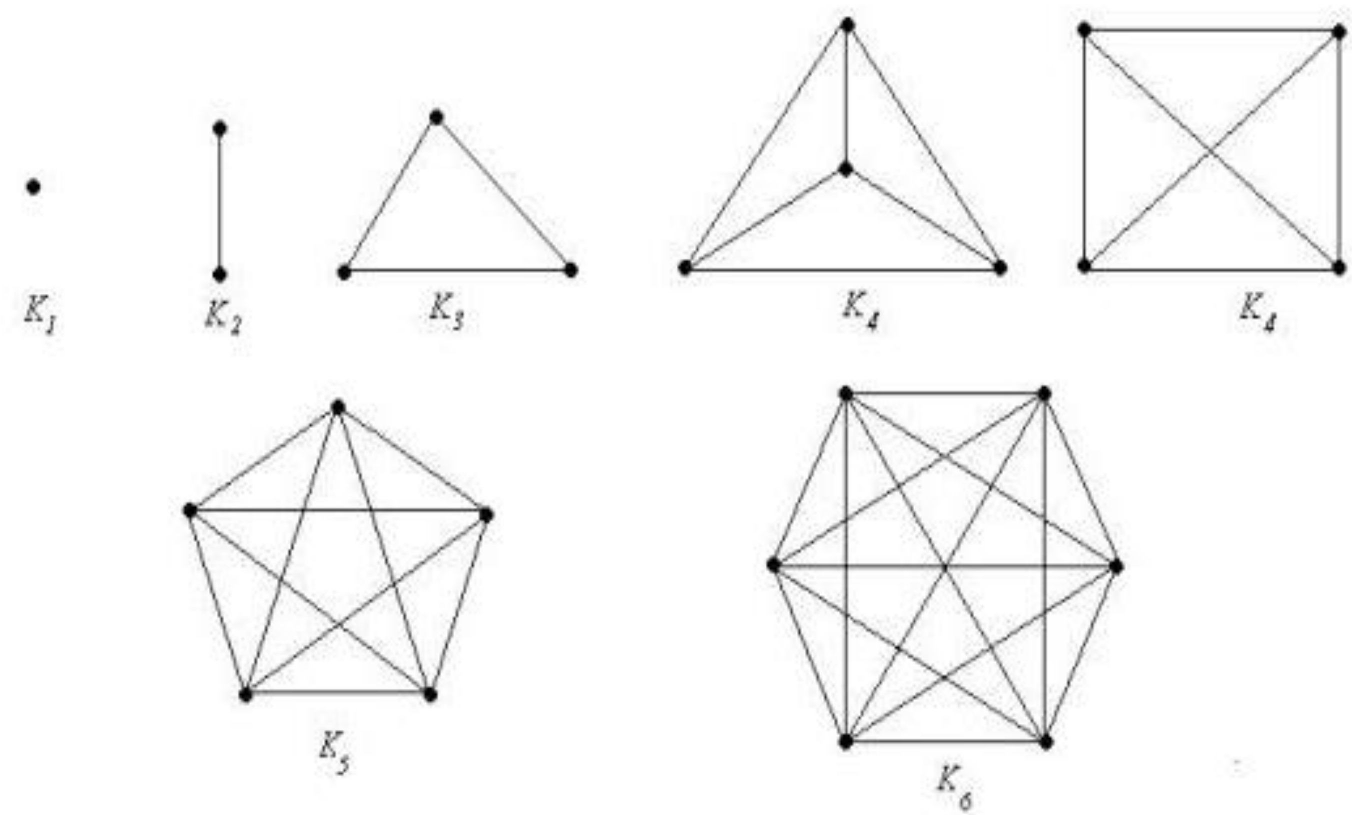


Figura 14. Ejemplo de grafos completos

<u>Grafo complementario.</u> El grafo complementario G' del grafo original G, está compuesto por todos los vértices de G y las aristas que no están en G.

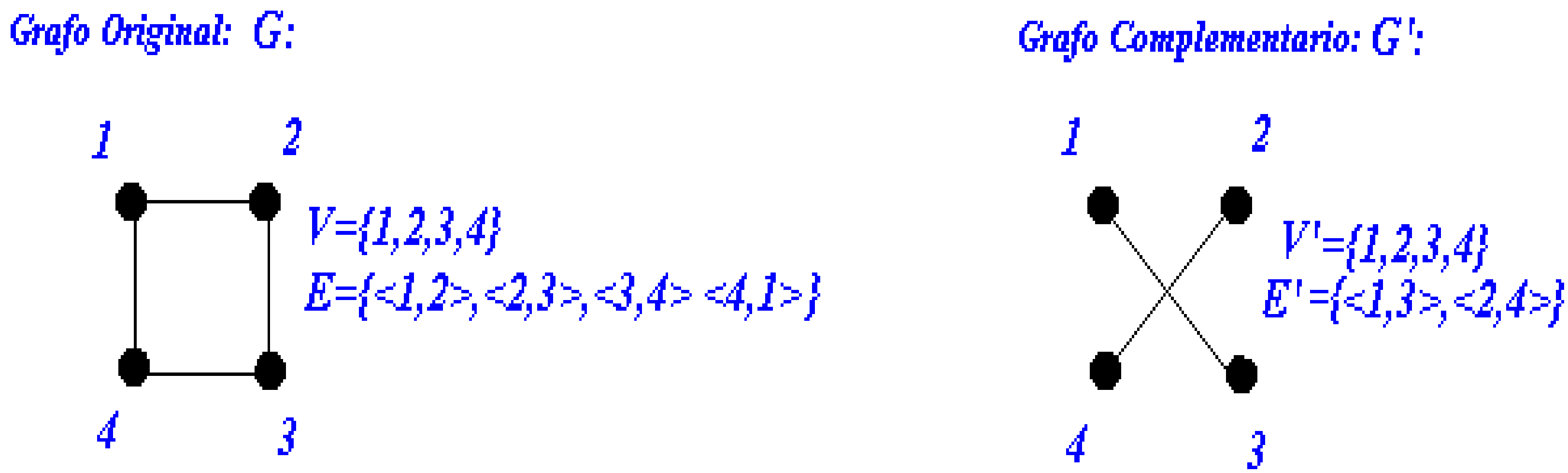


Figura 15. Ejemplo de grafo complementario

8. Isomorfismos. En esta sección se revisarán esos conceptos.

Grafos isomorfos. Dos grafos son isomorfos si y solo si para alguna ordenación de vértices y aristas sus matrices de incidencia son iguales.

Los pares de grafos de la figura 16 son isomorfos. Pueden comprobarlo dando un etiquetado en uno de los elementos de cada par y etiquetando de forma idéntica los vértices correspondientes del otro elemento del par.

Figura 16. Ejemplos de grafos isomorfos

Ejemplo 1. Sean los siguientes grafos denominados G1 y G2:

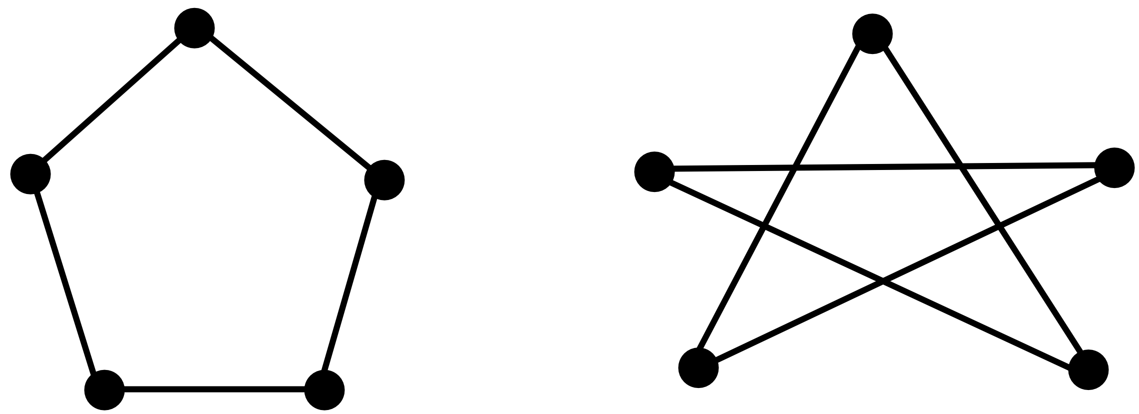


Figura 17-A. Grafos para verificar si son isomorfos

Etiquetamos el grafo de la izquierda:

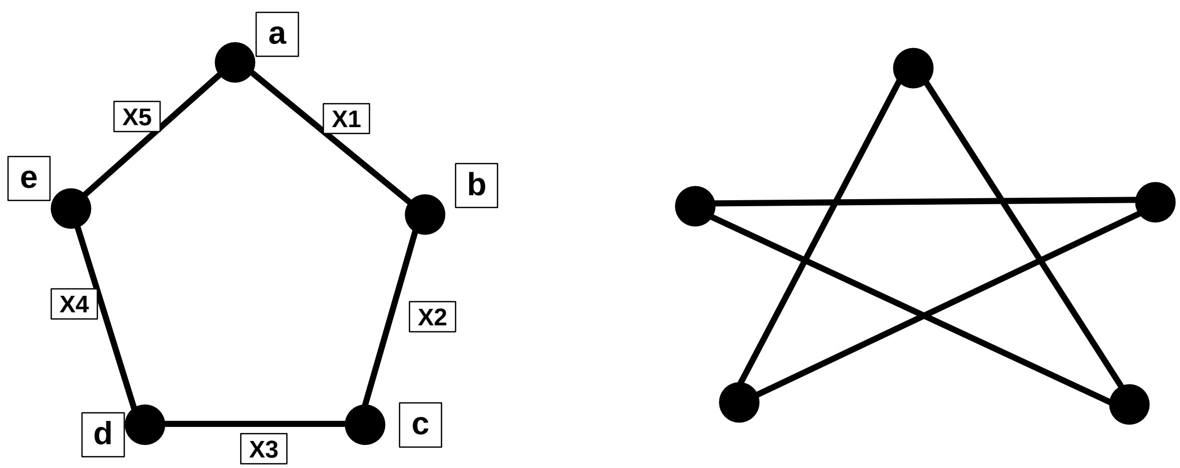


Figura 17-B. Verificación de isomorfismo. Paso 1.

Comenzamos con el vértice **a**: sus aristas X1, X5 que se mapean como Y1, Y5:
De las aristas mapeadas, relacionamos los vértices adyacentes de **a**, que son **e**, **b**.

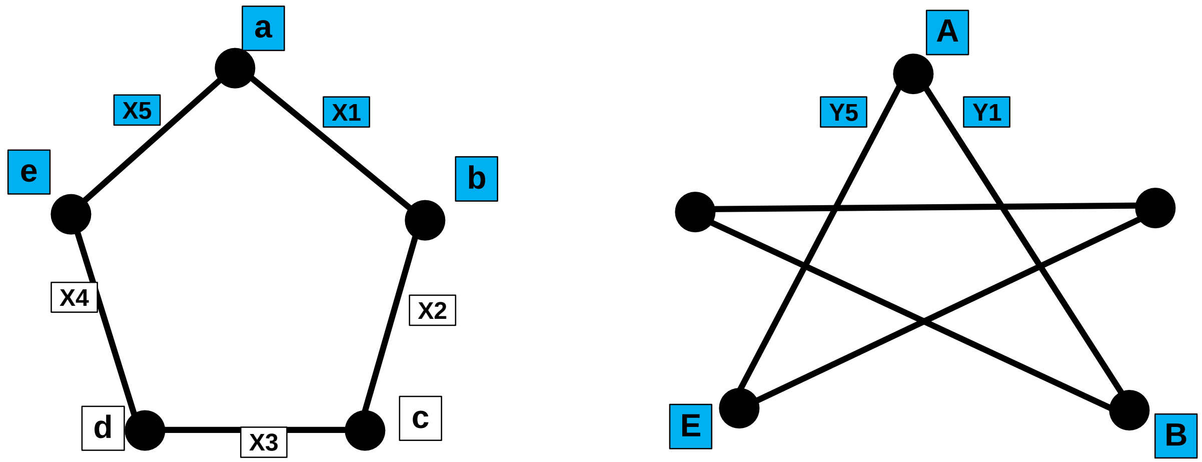


Figura 17-C. Verificación de isomorfismo. Paso 2.

Seguimos con el vértice **b**: sus adyacentes son **a** (ya mapeado como A) , **c**.
Las aristas son: **X1** (ya mapeada como Y1), **X2** .

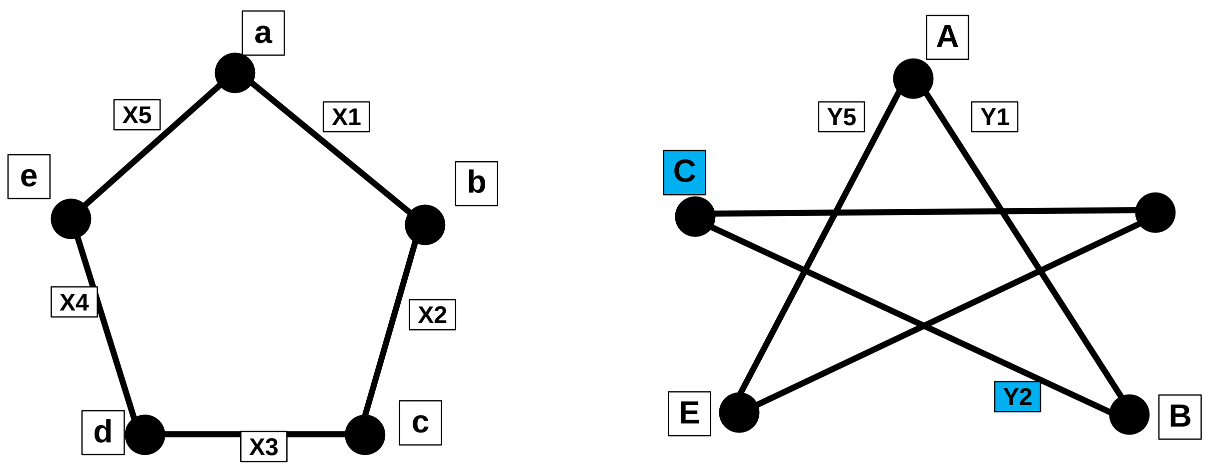


Figura 17-D. Verificación de isomorfismo. Paso 3.

Seguimos con el vértice **c**: sus adyacentes son **b** (ya mapeado como B) , **d**.
Las aristas son: **X2** (ya mapeada como Y2), **X3** .

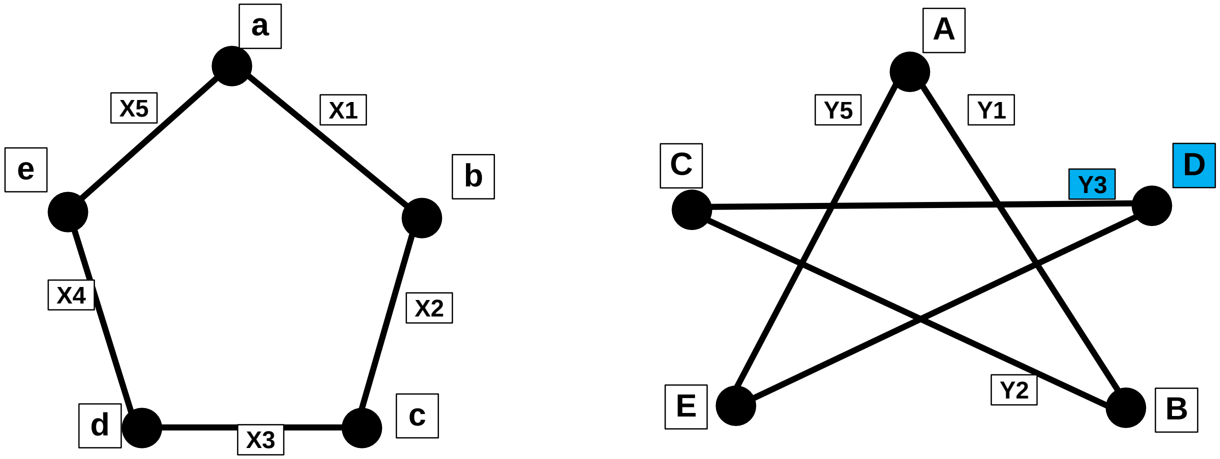


Figura 17-E. Verificación de isomorfismo. Paso 4.

Seguimos con el vértice **d**: sus adyacentes son **c** (ya mapeado como C) , **e**.
Las aristas son: **X3** (ya mapeada como Y3), **X4** .

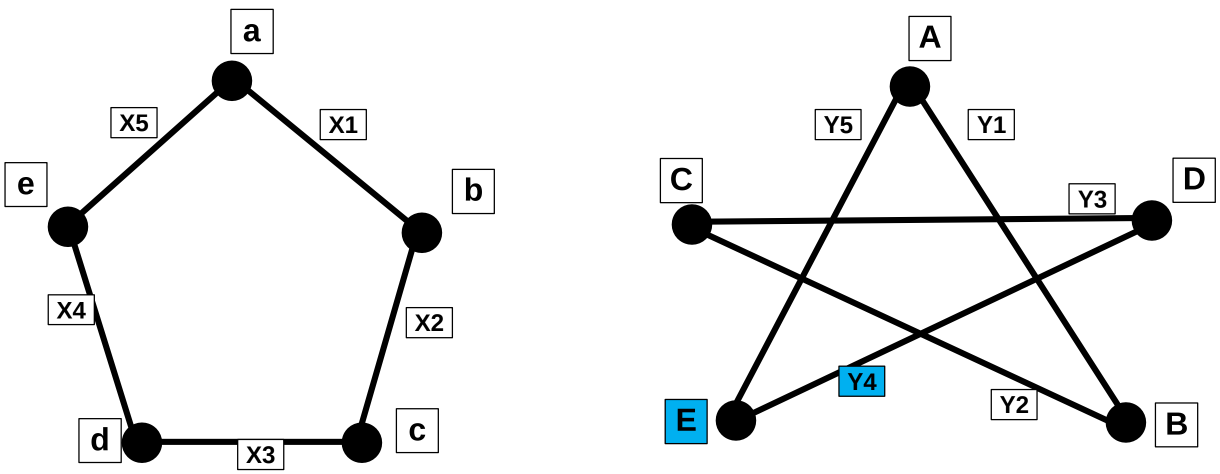


Figura 17-F. Verificación de isomorfismo. Paso 5.

Ya están mapeados todos los vértices y aristas.

y finalmente elaboramos las matrices de incidencia:

	X1	X2	X3	X4	X5
a	1				1
b	1	1			
c		1	1		
d			1	1	
e				1	1

	Y1	Y2	Y3	Y4	Y5
A	1				1
B	1	1			
C		1	1		
D			1	1	
E				1	1

Las matrices de incidencia son idénticas,
por consiguiente los grafos son isomorfos.

La verificación del isomorfismo es un problema difícil.

Para la resolución del problema se buscan datos necesariamente comunes a todos los grafos de una misma clase de isomorfía.

A estos datos se les llama invariantes de un grafo:

a) El número de vértices, |V|,

b) El número de aristas, |A|

c) la familia de los grados de los vértices, gr(x), para los $x \in V$

Que dos grafos tengan los mismos invariantes es una condición necesaria para que dos grafos sean isomorfos, pero no es una condición suficiente.

De hecho, no se conoce ningún conjunto de invariantes que sean suficientes para asegurar que dos grafos son isomorfos.

Ejemplo 2. En la figura siguiente, los grafos G1 , G2 y G3 tienen:

el mismo número de vértices,

el mismo número de aristas,

todos tienen 4 vértices de grado 2 y

tienen 2 vértices de grado 3;

pero NO son isomorfos.

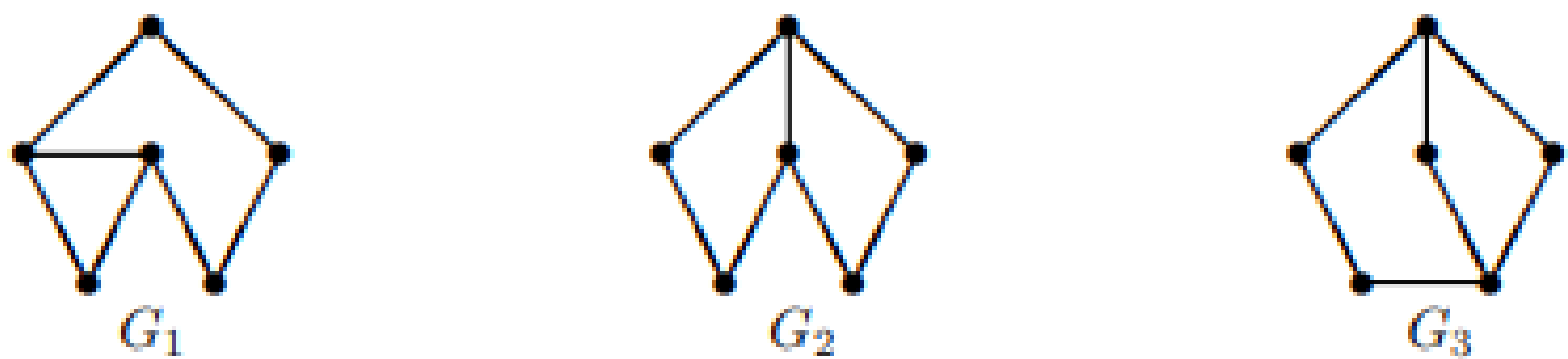


Figura 18. Grafos con invariantes comunes pero que NO SON isomorfos

Entonces, debemos comprobarlo.

COMPAREMOS G1 CON G2:

Etiquetamos el grafo de la izquierda

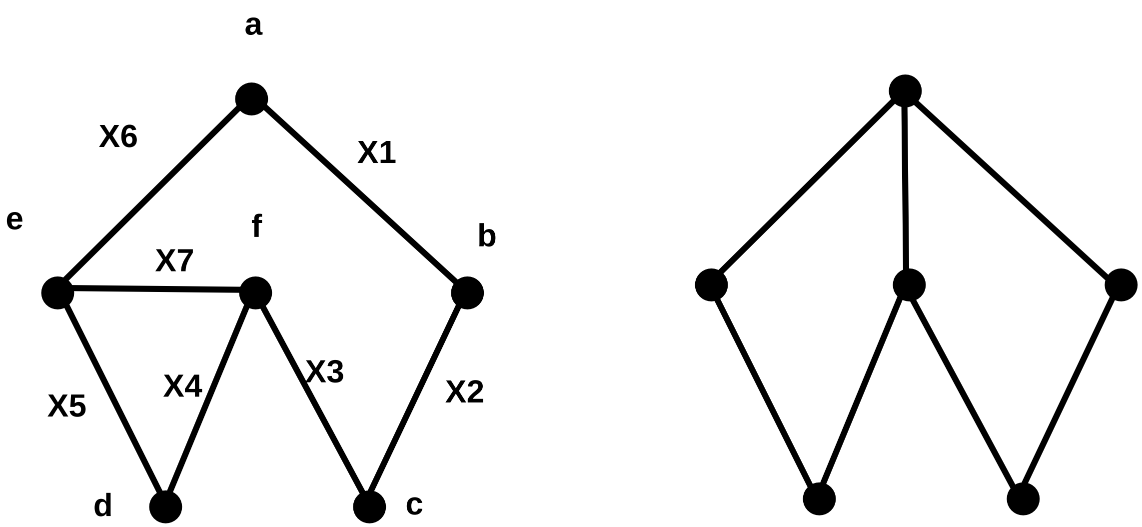


Figura 18.1-A.
Verificación de isomorfismo.
Paso 1.

Para el “mapeo” comenzamos con los vértices de mayor grado, que son:

el vértice **e** , cuyas aristas son X5, X6, X7

el vértice **f** , cuyas aristas son X3, X4, X7

Y buscamos esos vértices de grado 3 en este segundo grafo:

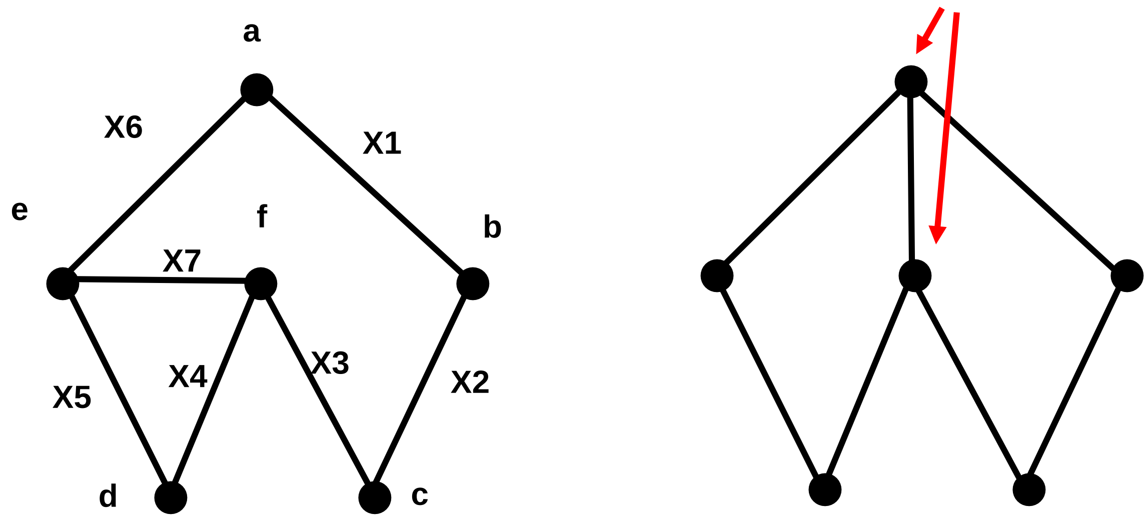


Figura 18.1-B.
Verificación de isomorfismo.
Paso 2.

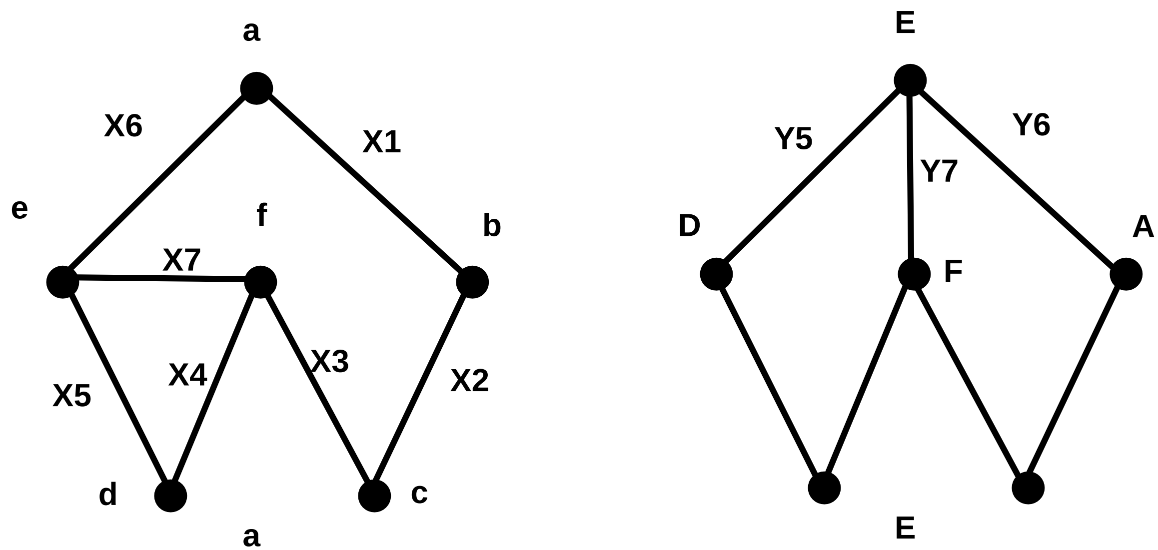


Figura 18.1-C.
Verificación de isomorfismo.
Paso 3.

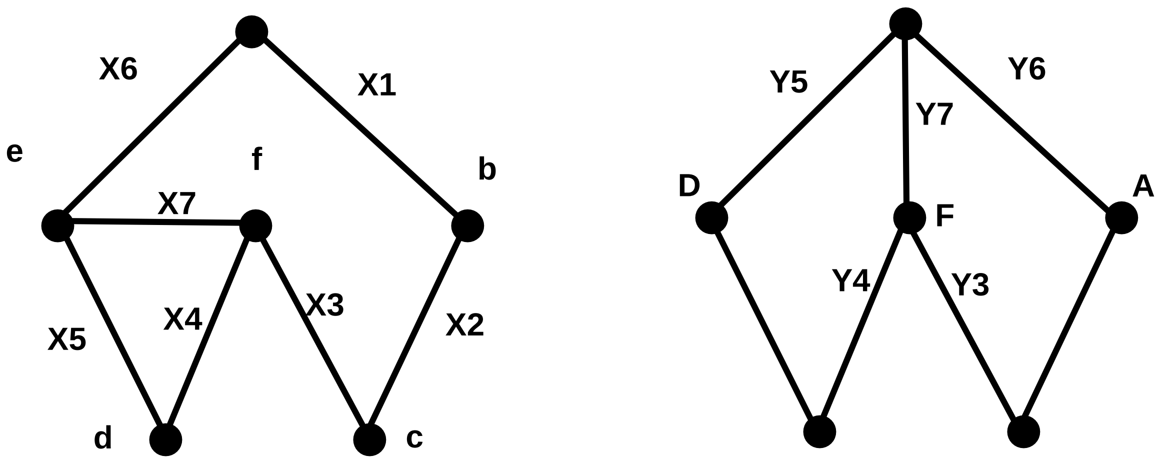


Figura 18.1-D.
Verificación de isomorfismo.
Paso 4.

En el grafo de la izquierda, el vértice a conecta con el vértice b, y ese vértice b, conecta con el vértice c, que finalmente conecta con el vértice f.

Es decir existe el camino <a, b, c, f > el cual NO tiene correspondencia con el grafo de la derecha.

AHORA COMPAREMOS G1 CON G3:

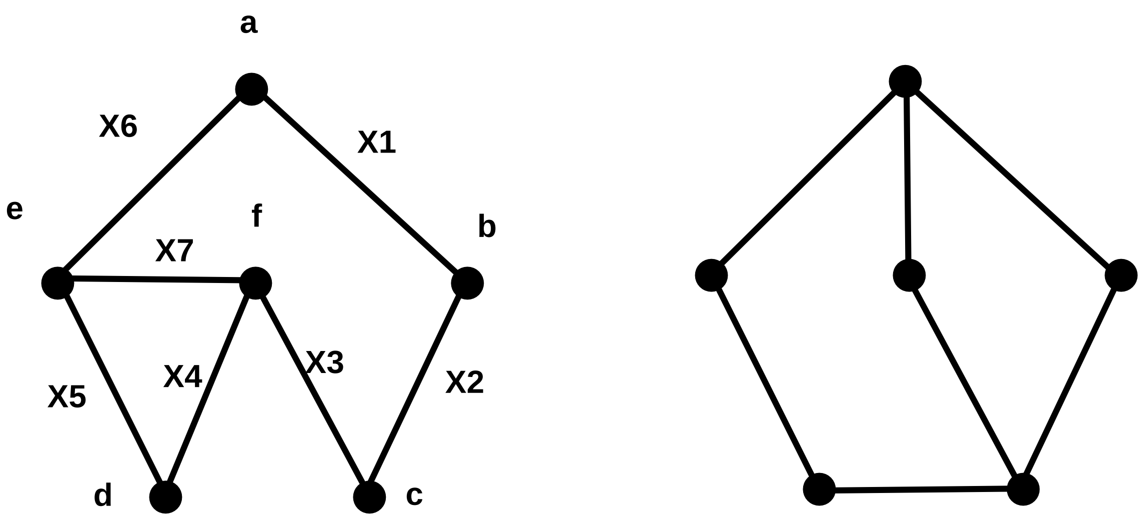


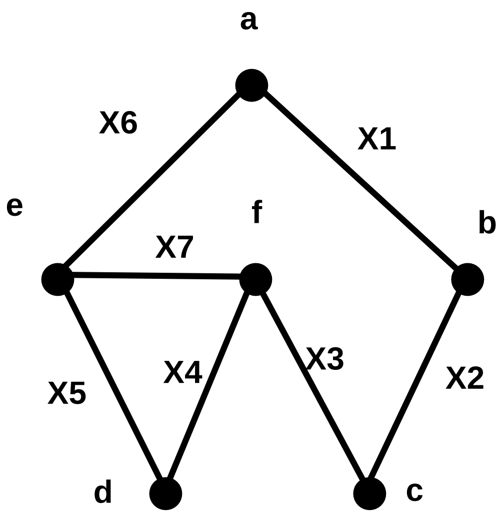
Figura 18.2-A.
Verificación de isomorfismo.
Paso 1.

Para el “mapeo” comenzamos con los vértices de mayor grado, que son:

el vértice **e** , cuyas aristas son X5, X6, X7

el vértice **f** , cuyas aristas son X3, X4, X7

Y buscamos esos vértices de grado 3 en este grafo G3:



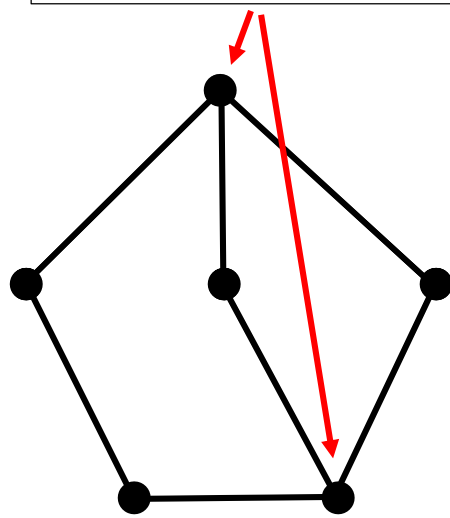
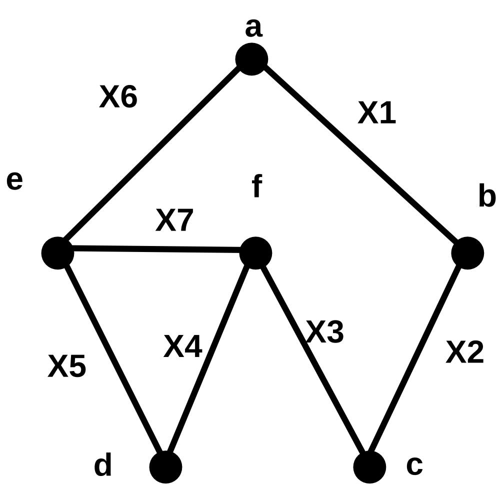
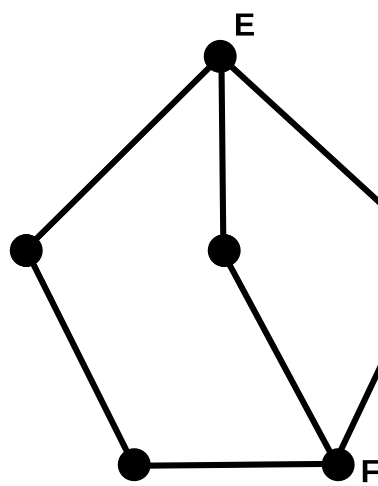


Figura 18.2-B.
Verificación de isomorfismo.
Paso 2.





Se aprecia que no está la arista que debería unir a los vértices E y F.

Por consiguiente, este proceso termina, y se comprueba que G1 y G3 NO son isomorfos.

Finalmente, comparemos los grafos G2 y G3:

Etiquetamos el grafo de la izquierda:

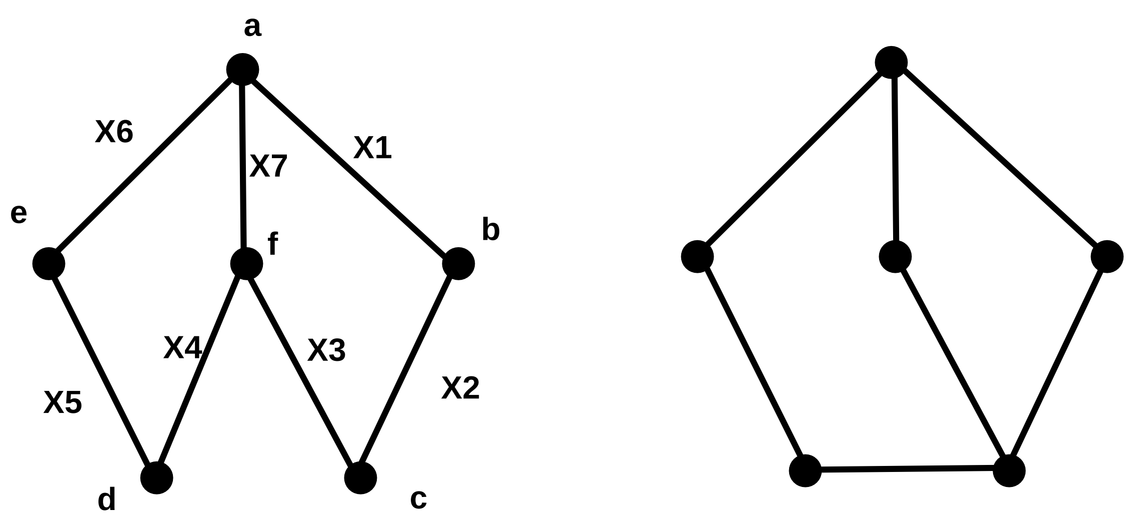


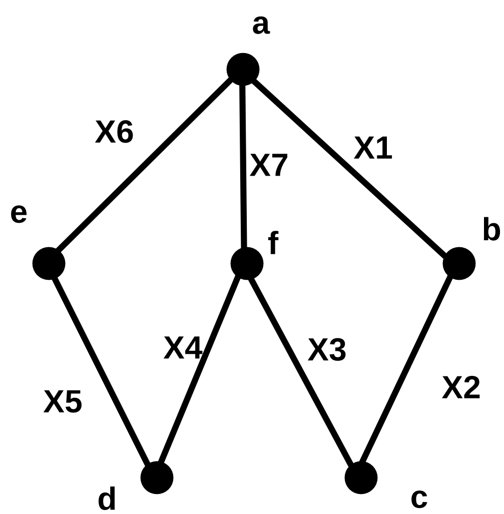
Figura 18.3-A.
Verificación de isomorfismo.
Paso 1.

Para el “mapeo” comenzamos con los vértices de mayor grado, que son:

el vértice **a** , cuyas aristas son X1, X6, X7

el vértice **f** , cuyas aristas son X3, X4, X7

Y buscamos esos vértices de grado 3 en G3:



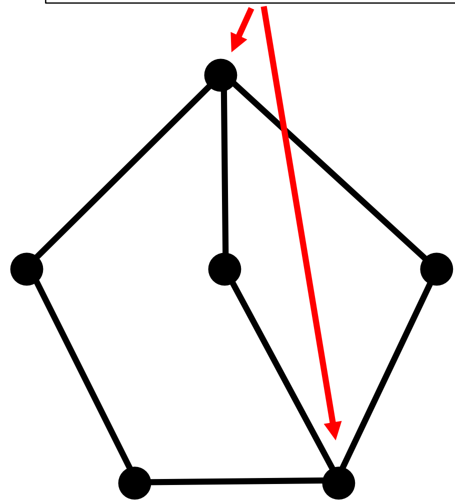
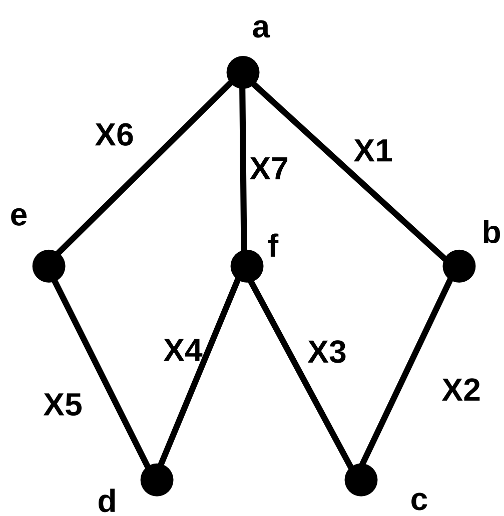
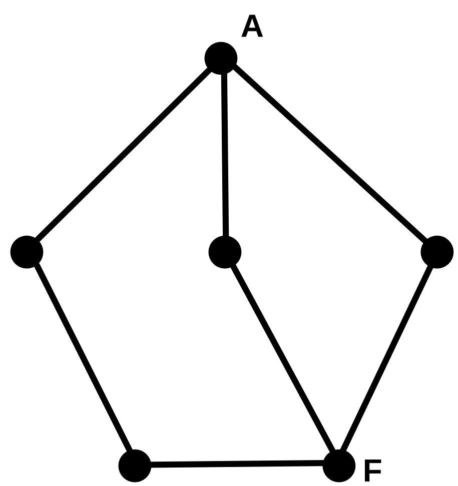


Figura 18.3-B.
Verificación de isomorfismo.
Paso 2.



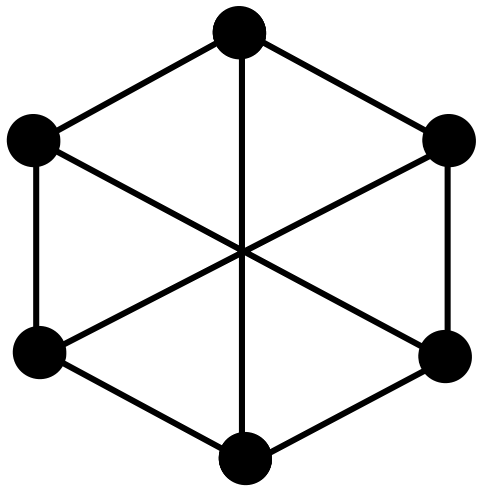


Se aprecia que NO está la arista que debería unir los vértices A y F.

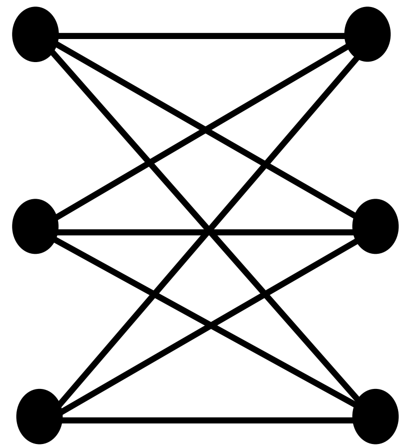
Por consiguiente, este proceso termina, y se comprueba que G2 y G3 NO son isomorfos.

Ejemplo 3. 6 amigos salen de vacaciones al mismo tiempo pero cada uno va a un lugar diferente. Antes de salir, deciden que al llegar a su destino cada uno de ellos enviará una postal a tres de esos amigos.

Entonces varios de ellos se plantean si es posible que cada amigo pueda recibir postales de precisamente los tres amigos a los que él les envió las suyas. Y proponen 2 grafos que representan la solución:



Grafo 1



Grafo 2

Figura 19. Grafos con invariantes comunes

La pregunta es: ¿Existe isomorfismo entre el El Grafo 1 y el Grafo 2?

Si lo hay indica que ambas alternativas son correctas al problema del envío de las postales.

Ambos grafos tienen: 6 vértices, 9 aristas, y los 6 vértices son de grado 3.

Etiquetamos el grafo de la izquierda:

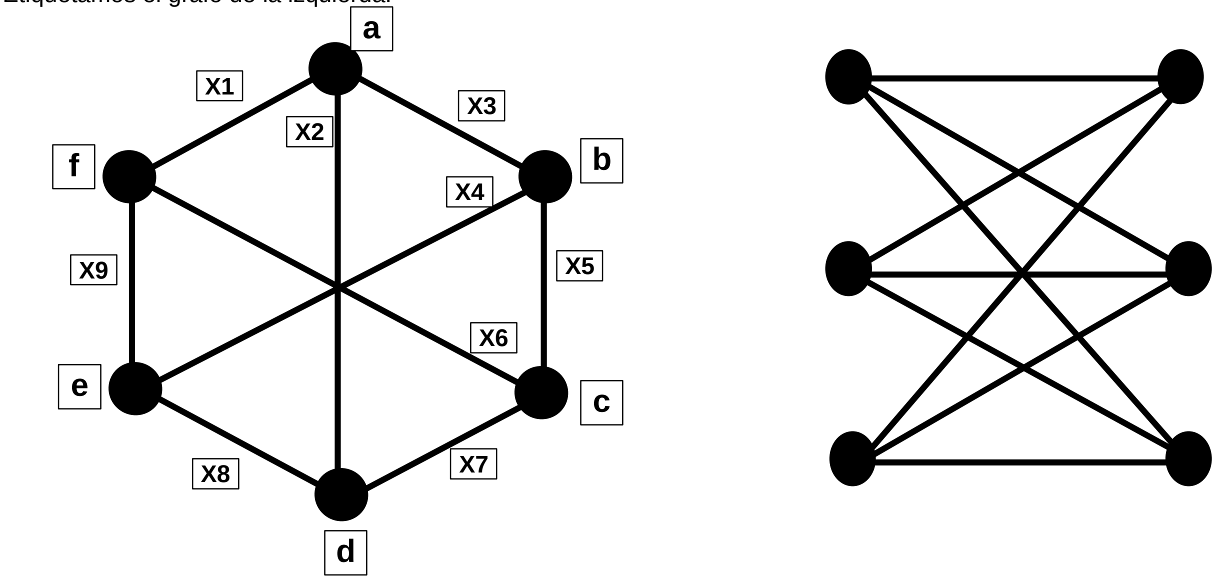


Figura 19.1.
Verificación de isomorfismo.
Paso 1.

Comenzamos con el vértice **a**: sus aristas X1, X2, X3 que se mapean como Y1, Y2, Y3:

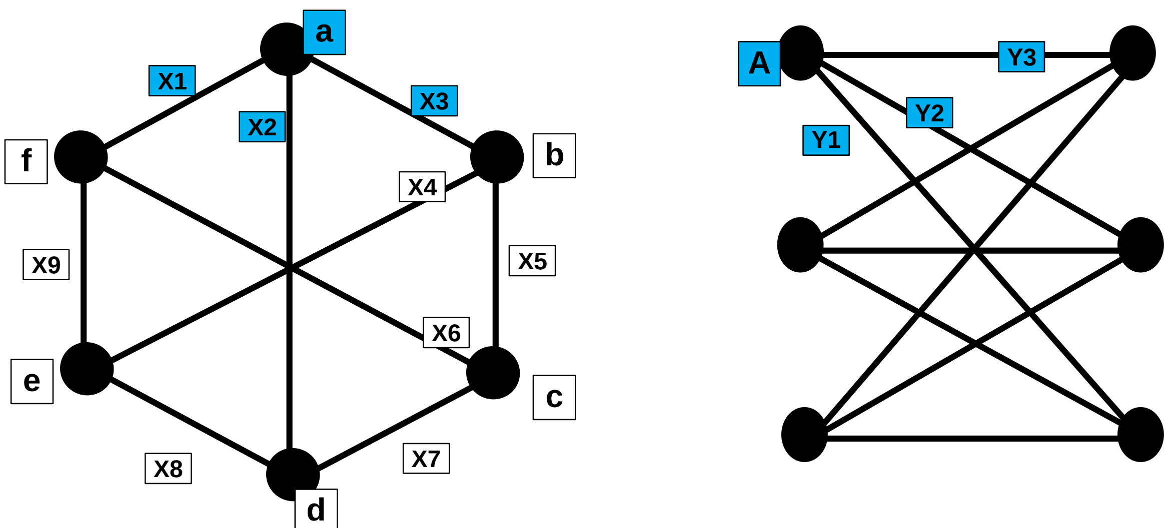


Figura 19.2.
Verificación de isomorfismo.
Paso 2.

De las aristas mapeadas, relacionamos los vértices adyacentes de **a**, que son **f** , **d**, **b**.

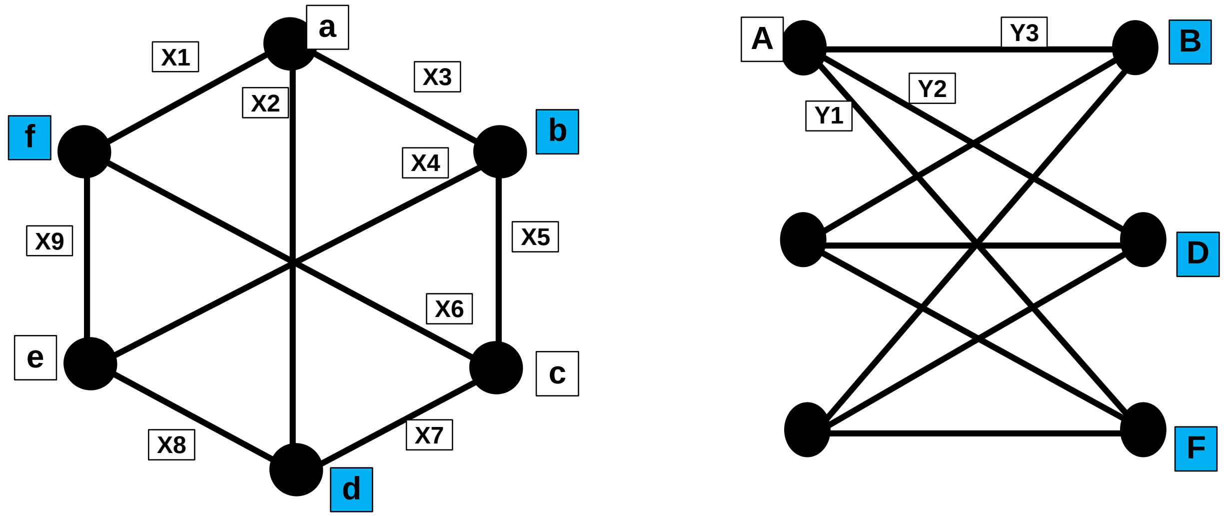


Figura 19.3.
Verificación de isomorfismo.
Paso 3.

Seguimos con el vértice **b**: sus adyacentes son **a** (ya mapeado como A) , **e**, **c**.
Las aristas son: **X3** (ya mapeada como Y3), **X4** y **X5**.

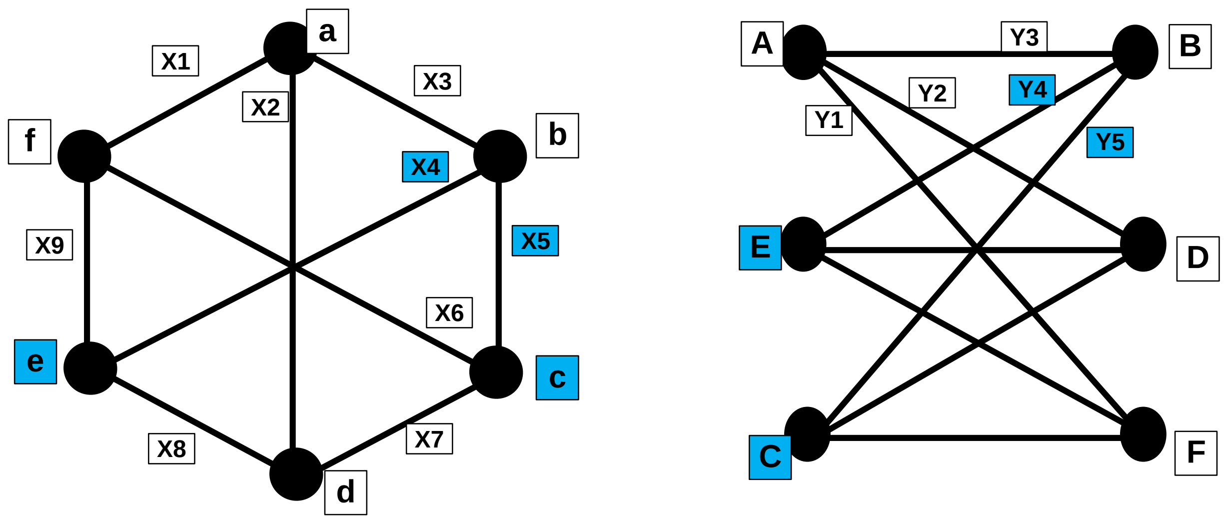


Figura 19.4.
Verificación de isomorfismo.
Paso 4.

Seguimos con el vértice **c**: sus adyacentes son **b**, **f**, **d** (ya mapeados).
Las aristas son: **X5** (ya mapeada como Y5), **X6** y **X7**.

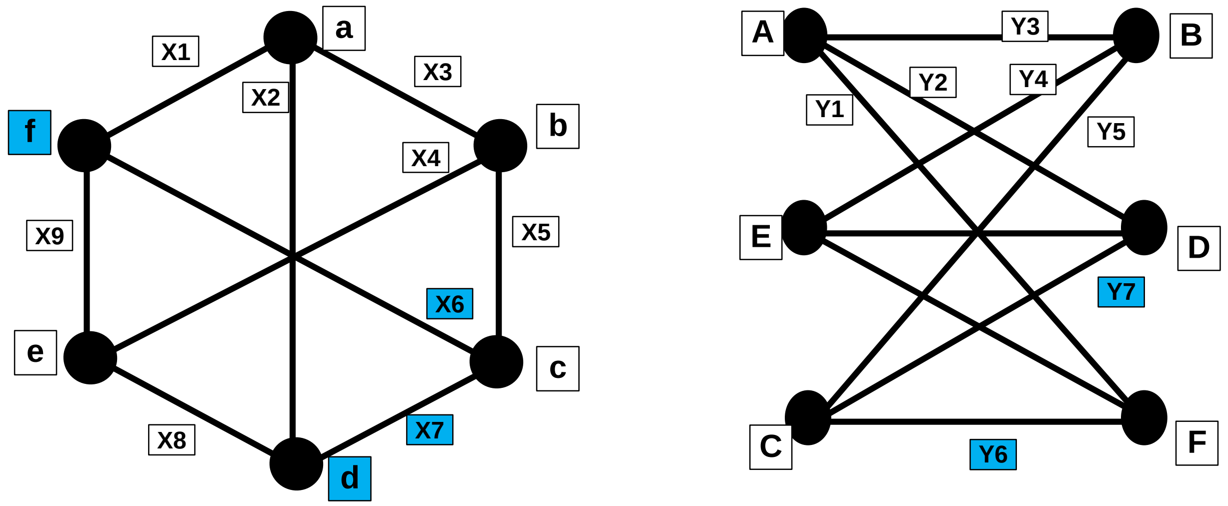


Figura 19.5.
Verificación de isomorfismo.
Paso 5.

Seguimos con el vértice **d**: sus adyacentes son **a** , **c**, **e** (ya mapeados).
Las aristas son: **X2** (ya mapeada como Y2), **X7** (ya mapeada como Y7) y **X8**.

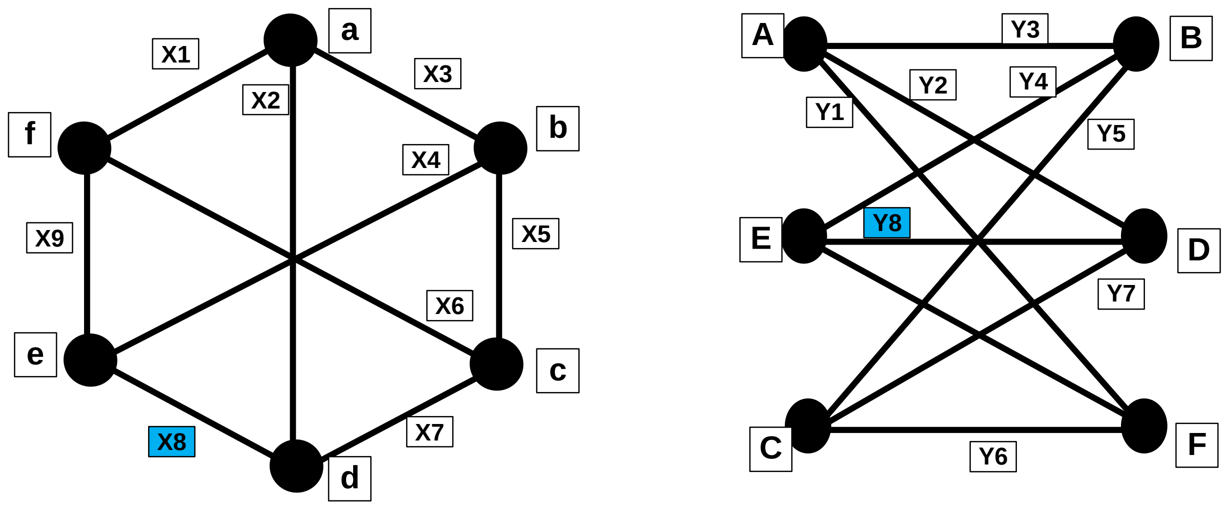
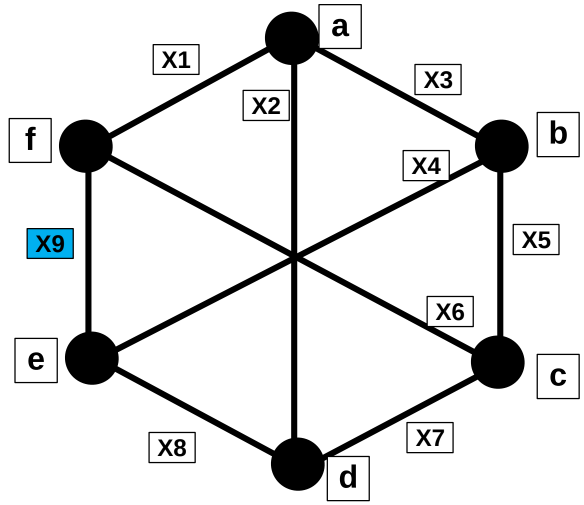


Figura 19.6.
Verificación de isomorfismo.
Paso 6.

Seguimos con el vértice **e**: sus adyacentes son **b** , **d**, **f** (ya mapeados).
Las aristas son: **X4** (ya mapeada como Y4), **X8** (ya mapeada como Y8) y **X9**



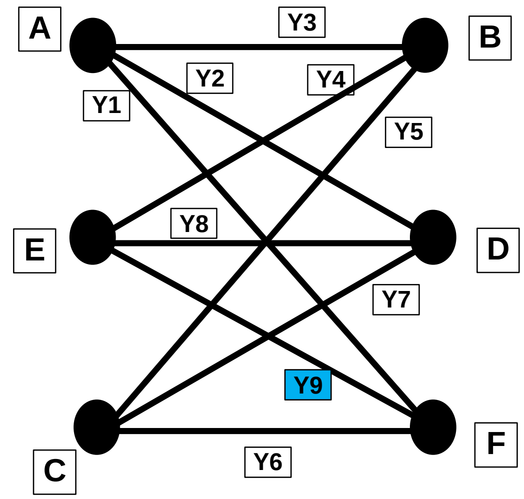


Figura 19.7.
Verificación de isomorfismo.
Paso 7.

Ya están mapeados todos los vértices y aristas.
y finalmente elaboramos las matrices de incidencia:

	X1	X2	X3	X4	X5	X6	X7	X8	X9
a	1	1	1						
b			1	1	1				
c					1	1	1		
d		1					1	1	
e				1				1	1
f	1					1			1

	Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8	Y9
A	1	1	1						
B			1	1	1				
C					1	1	1		
D		1					1	1	
E				1				1	1
F	1					1			1

Las matrices de incidencia son idénticas,

por consiguiente los grafos son isomorfos.

PARTE 2. RECORRIDOS DE GRAFOS.

El concepto de recorrido del grafo consiste en visitar todos los vértices del grafo sucesivamente de manera sistemática de manera que cada vértice se visite ***una única vez***.

En el recorrido de un grafo, existen dos tipos de vértices:

a) *Vértices visitados:* vértices ya visitados en el recorrido

b) *Vértices frontera:* vértices que aún no ha sido visitados pero están conectados con algún vértice visitado (están pendientes de visitar).

Al igual que un árbol (de hecho, un árbol es un tipo de grafo orientado sin ciclos), un grafo puede ser recorrido en profundidad o en amplitud.

Existen dos diferencias fundamentales a la hora de recorrer un grafo respecto de un árbol:

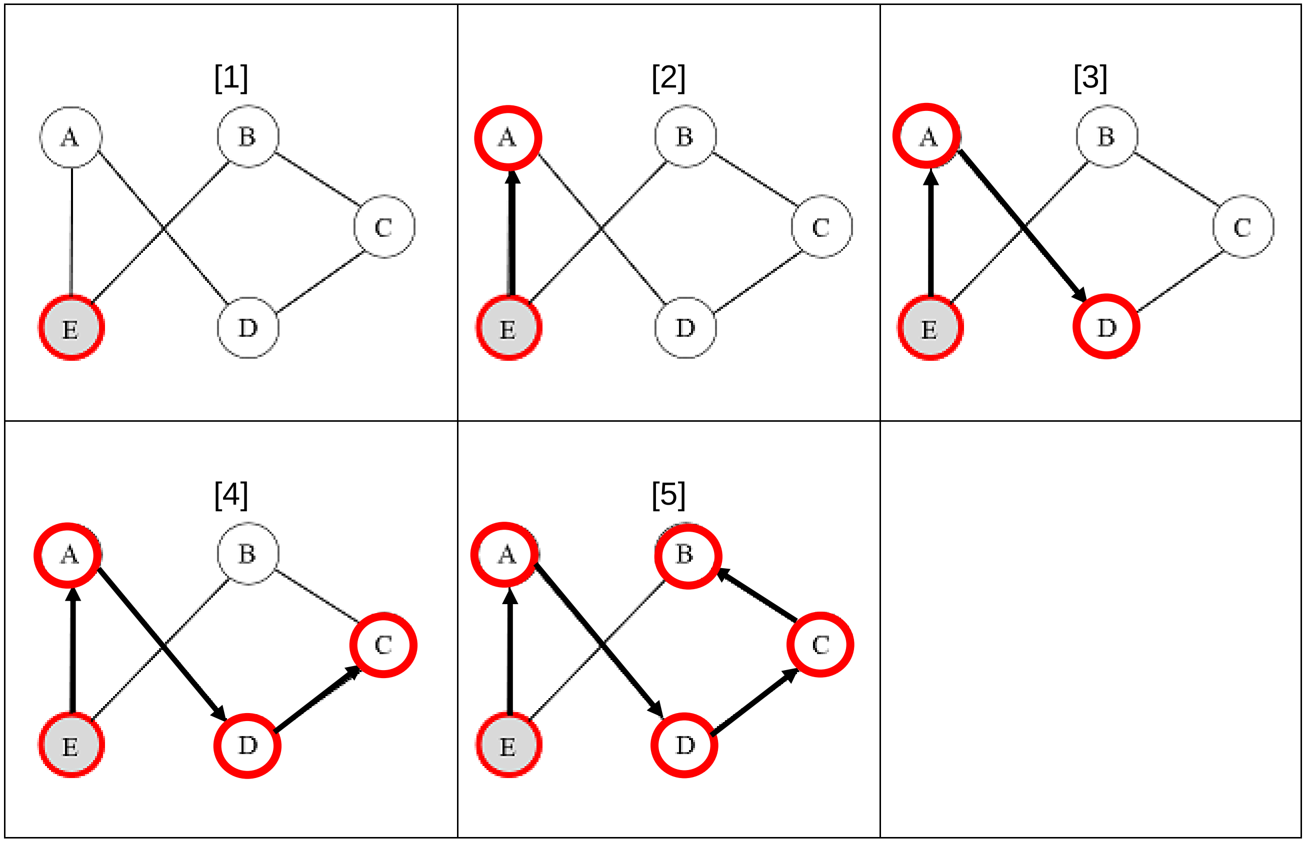
a) Puesto que un árbol es un grafo dirigido acíciclo, al avanzar en el recorrido no cabe la posibilidad de que se vuelva a visitar un vértice ya visitado. En el recorrido de un grafo sí cabe la posibilidad de al avanzar visitar un vértice ya visitado, de modo que debe evitarse esta situación.

b) Partiendo de la raíz de un árbol se pueden visitar todos los vértices, mientras que en un grafo se puede dar la posibilidad de que no se alcancen todos los vértices desde un vértice. Habría que comenzar el recorrido en otro vértice para alcanzar todos los vértices.

Cabe anotar que como ***no hay un nodo raíz***, es preciso fijar (de manera arbitraria ó según la situación) un origen para el recorrido.

1. Recorrido en profundidad (Depth-First Search –DFS-). A partir del nodo elegido como origen, avanzar a otro nodo no visitado y seguir el recorrido, de la misma manera, a partir de él.

Ejemplo 1. Recorrido en profundidad en un grafo no dirigido.

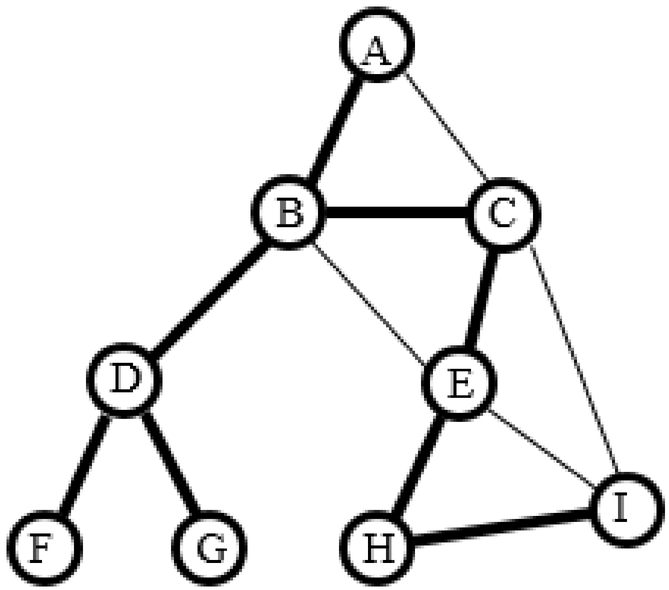


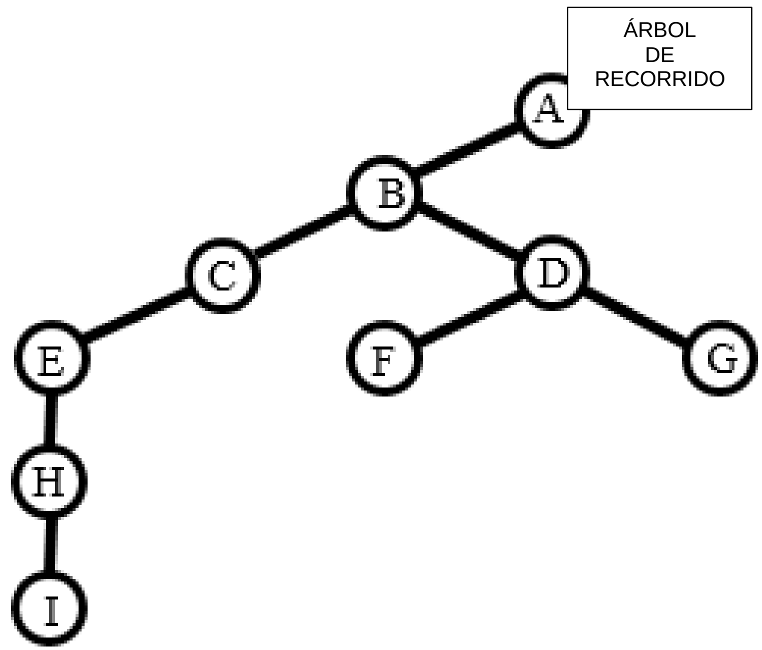
Más formalmente se especifica el recorrido de la siguiente manera:

Se puede tener una estructura como un arreglo en donde se indique que cada nodo del grafo está sin visitar, así se podrá llevar un seguimiento de cuáles nodos han sido visitados.

```
Se toma como vértice inicial E,
E ← visitado (el conjunto de vértices adyacentes a  E  son  A , B)

Se toma vértice A
A ← visitado (el conjunto de vértices adyacentes a  A  son  E , D)
//E ya está visitado, por lo que se continúa con D

Se toma vértice D
D ← visitado (el conjunto de vértices adyacentes a  D  es  C)

Se toma vértice C
C ← visitado (el conjunto de vértices adyacentes a  C  son  B , D)
//D ya está visitado, por lo que se continúa con B

Se toma vértice B
B ← visitado (el conjunto de vértices adyacentes a  B  es  E)
//E ya está visitado, no hay más vértices, por lo que se termina el recorrido
```

Ejemplo 2. Efectuar un recorrido en profundidad del grafo tomando como inicial el nodo A.





Se toma como vértice inicial A,
A ← visitado (el conjunto de vértices adyacentes a A son B, C)

Se toma vértice B
B ← visitado (el conjunto de vértices adyacentes a B son A, C, D, E)
//A ya está visitado, por lo que se continúa con C

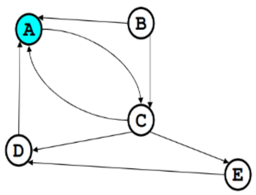
Se toma vértice C
C ← visitado (el conjunto de vértices adyacentes a C son A, B, E, I)
//A , B ya están visitados, por lo que se continúa con E

Se toma vértice E
E ← visitado (el conjunto de vértices adyacentes a E son B, C, H, I)
//B , C ya están visitados, por lo que se continúa con H

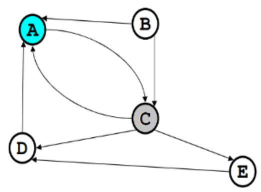
```
Se toma vértice H
H ← visitado (el conjunto de vértices adyacentes a  H  son  E, I)
//E ya está visitado, por lo que se continúa con I

Se toma vértice I
I ← visitado (el conjunto de vértices adyacentes a  I  son  C, E, H)
//C, E, H ya están visitados, se termina el recorrido en profundidad a partir de I

//Se hace un "backtracking" desde I y revisar sus adyacencias, y al retroceder hasta el nodo B, aún
//falta el nodo D, de modo que se puede continuar el recorrido

Se toma vértice D
D ← visitado (el conjunto de vértices adyacentes a  D  son  F, G)
//B ya está visitado, por lo que se continúa con F

Se toma vértice F
F ← visitado (el conjunto de vértices adyacentes a  F  es D)
//D ya está visitado, se termina el recorrido en profundidad a partir de F

//Se hace "backtracking" desde F y revisar sus adyacencias, y al retroceder hasta el nodo D, aún falta
//el nodo G, de modo que se puede continuar el recorrido

Se toma vértice G
G ← visitado (el conjunto de vértices adyacentes a  G  es D)
//D ya está visitado, no hay más vértices, por lo que se termina el recorrido
```

Ejemplo 3. Recorrido en profundidad en un grafo dirigido iniciando en el nodo A.

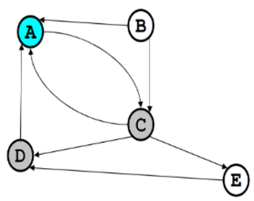
[1]



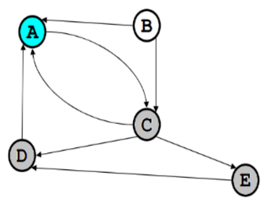
[2]



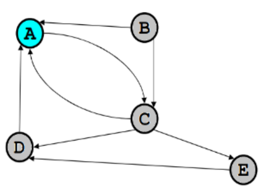
[3]



[4]



[5]



Se toma como vértice inicial A,
A ← visitado (el conjunto de vértices adyacentes desde A es C)

Se toma vértice C
C ← visitado (el conjunto de vértices adyacentes desde C son A, D, E)
//A ya está visitado, por lo que se continúa con D

Se toma vértice D
D ← visitado (el conjunto de vértices adyacentes desde D es A)
//A ya está visitado, se termina el recorrido en profundidad a partir de D

//Se hace "backtracking" desde D y revisar sus adyacencias, y al retroceder
//hasta el nodo C, aún falta el nodo E, de modo que se puede continuar el
//recorrido

Se toma vértice E
E ← visitado (el conjunto de vértices adyacentes desde E es D)
//D ya está visitado, se termina el recorrido en profundidad a partir de E

No es posible alcanzar más vértices desde A, de manera que hay que seleccionar un nuevo vértice desde el que recomenzar la exploración en profundidad, se elige el vértice faltante

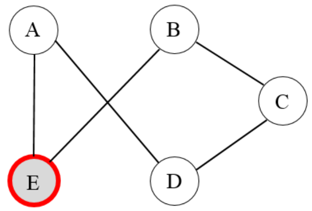
Se toma vértice B
B ← visitado (el conjunto de vértices adyacentes desde B son A , C)
//A, C ya están visitados de manera que se termina el recorrido en
//profundidad a partir del vértice B

No hay más vértices, por lo que se termina el recorrido.

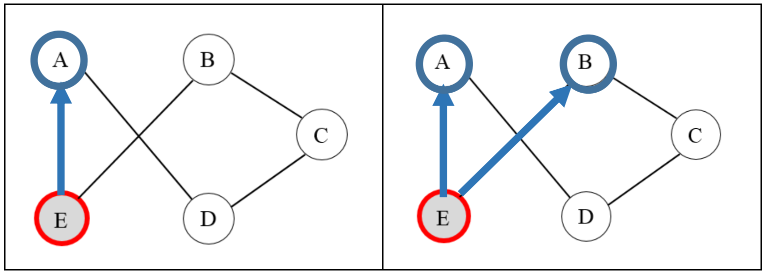
2. Recorrido en amplitud ó anchura (Breadth-First Search –BFS-). En un recorrido en amplitud, se elige un vértice no visitado v, como punto de partida y se pasa a visitar cada uno de sus vértices adyacentes, para continuar posteriormente visitando los adyacentes a estos últimos y así sucesivamente hasta que no se puedan alcanzar más vértices. Si queda algún vértice sin visitar, se selecciona y se vuelve a relanzar el proceso.

Ejemplo 4. Recorrido en amplitud en un grafo no dirigido.

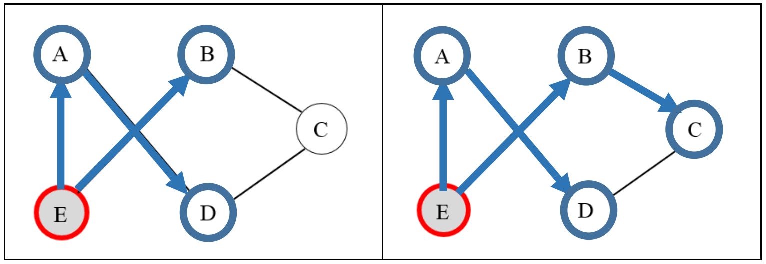
Etapa 1. Se toma el nodo de origen



Etapa 2. Acceder a todos los nodos que están a distancia 1 del nodo origen, es decir, directamente relacionados con el origen (existe un arco que los une).



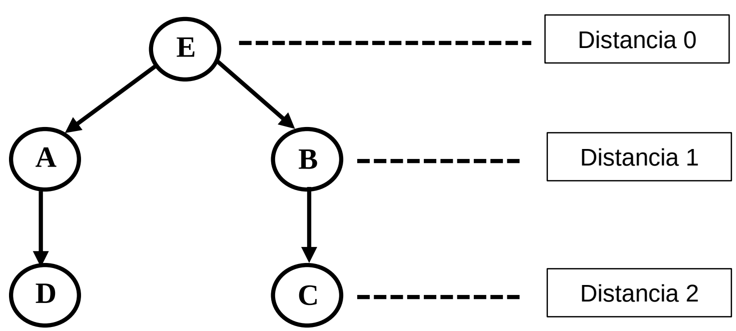
Etapa 3. Acceder a todos los nodos que están a distancia 2, es decir, directamente relacionados con los que estaban a distancia 1 en la Etapa anterior. Y así hasta cubrir los nodos.



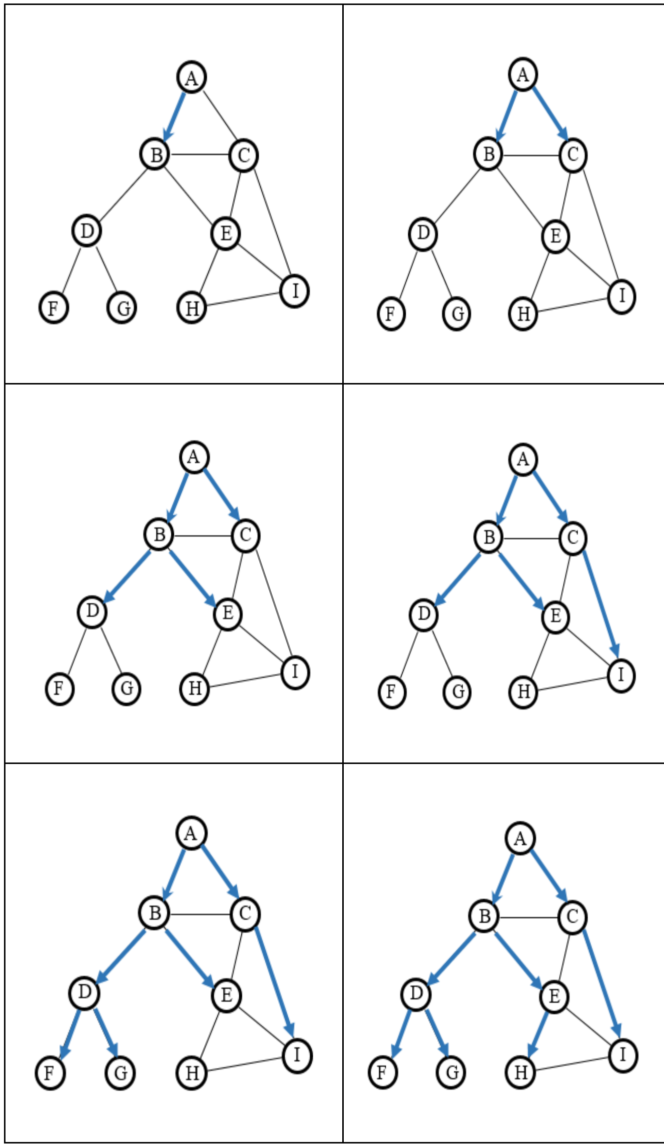
Más formalmente se especifica el recorrido de la siguiente manera:

```
Se toma como vértice inicial E,
E ← visitado (el conjunto de vértices adyacentes a  E  son  A, B)

Se toma vértice A
A ← visitado (el conjunto de vértices adyacentes a  A  son  E, D)
//E ya está visitado. Backtracking al nodo E y se revisan adyacencias

Se toma vértice B
B ← visitado (el conjunto de vértices adyacentes a  B  son  C, E)
//E ya está visitado
//backtracking al nodo A y se revisan adyacencias

Se toma vértice D
D ← visitado (el conjunto de vértices adyacentes a  D  son  A, C)
//backtracking al nodo B y se revisan adyacencias

Se toma vértice C
C ← visitado (el conjunto de vértices adyacentes a  C  son  B, D)
//B, D ya están visitados, no hay más vértices, por lo que se termina el recorrido
```

El resultado del recorrido es un árbol, que incluye los nodos visitados y los arcos utilizados para acceder a ellos.



Ejemplo 5. Efectuar un recorrido en amplitud del grafo tomando como inicial el nodo A.



Se toma como vértice inicial A,
A ← **visitado** (el conjunto de vértices adyacentes a A son B, C)

Se toma vértice B
B ← **visitado** (el conjunto de vértices adyacentes a B son C, D, E)
//C ya está visitado. Backtracking al nodo A y se revisan adyacencias

Se toma vértice C
C ← **visitado** (el conjunto de vértices adyacentes a C son B, E, I)
//B, E ya están visitados

//backtracking al nodo B y se revisan adyacencias
Se toma vértice D
D ← **visitado** (el conjunto de vértices adyacentes a D son F, G)

//backtracking al nodo B y se revisan adyacencias
Se toma vértice E
E ← **visitado** (el conjunto de vértices adyacentes a E son C, H, I)
//C ya está visitado. Backtracking al nodo C y se revisan adyacencias

Se toma vértice I
I ← **visitado** (el conjunto de vértices adyacentes a I son C, E, H)
//C, E ya están visitados. Backtracking al nodo D y se revisan adyacencias

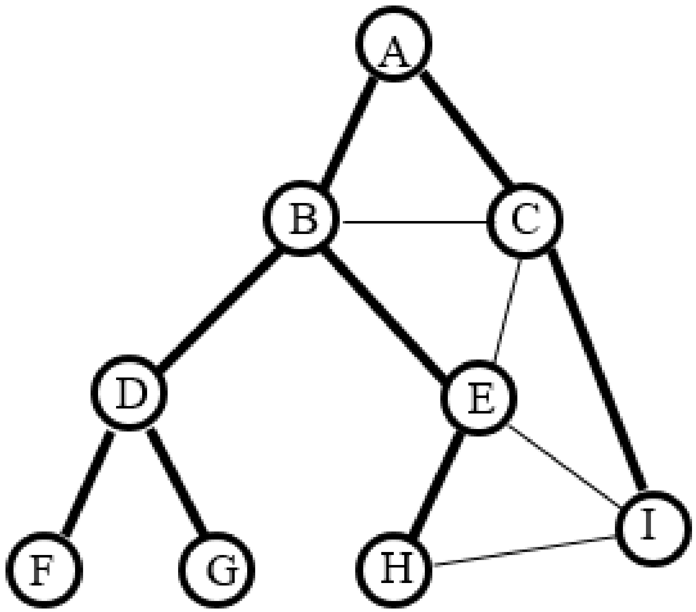
Se toma vértice F
F ← **visitado** (el conjunto de vértices adyacentes a F es D)
//D ya está visitado. Backtracking al nodo D y se revisan adyacencias

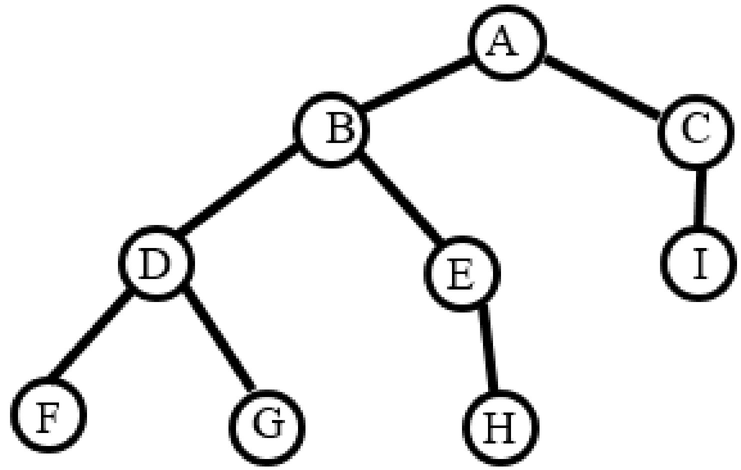
Se toma vértice G
G ← **visitado** (el conjunto de vértices adyacentes a G es D)
//D ya está visitado. Backtracking al nodo E y se revisan adyacencias

Se toma vértice H
H ← **visitado** (el conjunto de vértices adyacentes a H son E, I)
//E , I ya están visitados. No hay más vértices, por lo que se termina el
//recorrido

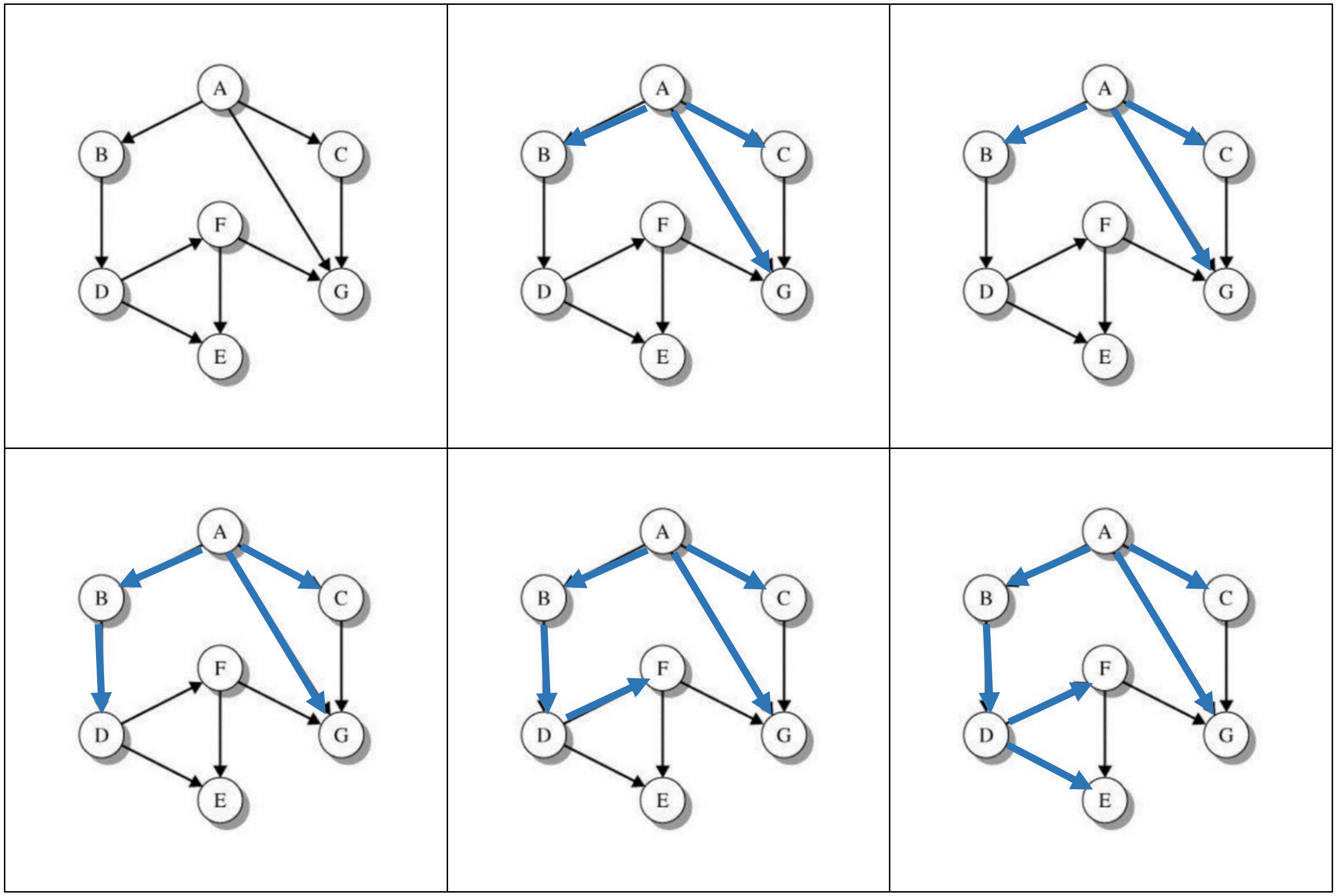
Resultado:

ÁRBOL DE RECORRIDO





Ejemplo 6. Recorrido en amplitud en un grafo dirigido iniciando en el nodo A.



```
Se toma como vértice inicial A,
A ← visitado (el conjunto de vértices adyacentes desde  A  son B, C, G)

Se toma vértice B
B ← visitado (el conjunto de vértices adyacentes desde  B  es D)
// Backtracking al nodo A y se revisan adyacencias

Se toma vértice C
C ← visitado (el conjunto de vértices adyacentes desde  C  es G)
// Backtracking al nodo A y se revisan adyacencias

Se toma vértice G
G ← visitado (el conjunto de vértices adyacentes desde  G  es vacío)
// Backtracking al nodo B y se revisan adyacencias

Se toma vértice D
D ← visitado (el conjunto de vértices adyacentes desde  D  son F, E)

// Backtracking al nodo C y se revisan adyacencias
//G ya está visitado

// Backtracking al nodo G y se revisan adyacencias
//G ya está visitado, no hay adyacencias.

//Baktracking nodo D y se revisan adyacencias

Se toma vértice F
F ← visitado (el conjunto de vértices adyacentes desde  F  es E, G)
//G ya visitado
//Baktracking nodo D y se revisan adyacencias

Se toma vértice E
E ← visitado (el conjunto de vértices adyacentes desde  F  es G)
//G ya visitado
//No hay más vértices, por lo que se termina el recorrido
```