

Introducción a Grafos

Estructuras de Datos

Andrea Rueda

Pontificia Universidad Javeriana
Departamento de Ingeniería de Sistemas

Grafos

Grafos

- Matemáticas discretas, teoría de grafos.

Representación de un conjunto de objetos (vértices) relacionados por conexiones (aristas).

- Estructura no lineal.
- Modelado de estrategias de:
 - Relaciones.
 - Discretización del espacio.
 - Caminos.
 - Geometría computacional.

Grafos

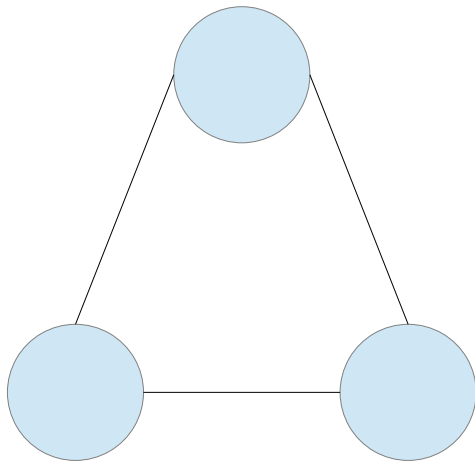
- Definición formal:

Par ordenado $G = (V, E)$.

- V : conjunto de vértices.
- E : conjunto de aristas \rightarrow representadas como un par (no) ordenado de los vértices conectados (x,y) .

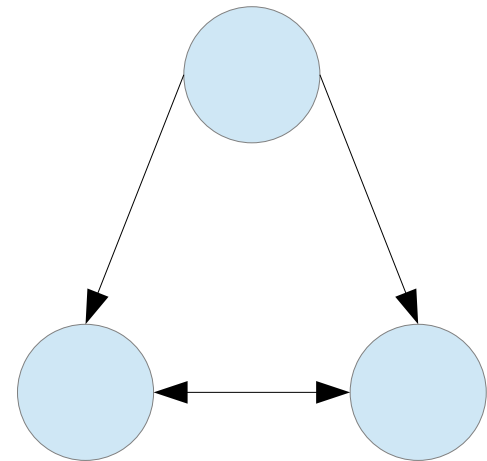
Tipos de grafos

- Grafos no dirigidos:



Aristas sin
orientación.

- Grafos dirigidos (digrafos):



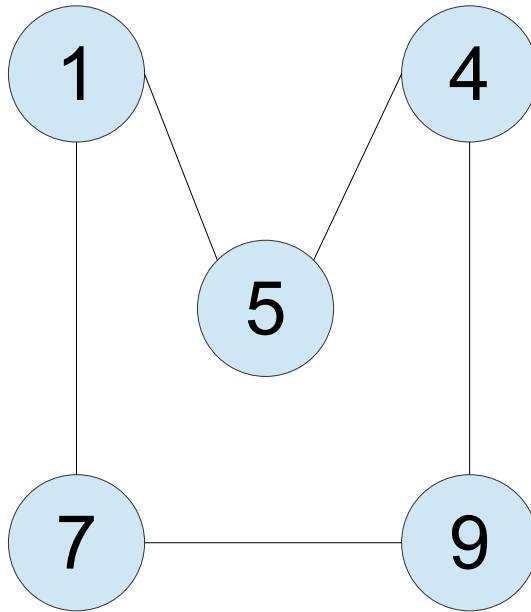
Aristas en un sólo
sentido (o en doble
sentido).

Tipos de grafos

- Grafos dirigidos (digrafos):
grafo donde las aristas son pares ordenados
 $(x,y) \neq (y,x)$.
- Grafos no dirigidos:
grafo donde las aristas son pares no ordenados
 $(x,y) = (y,x)$.

Grafos

- Ejemplo:

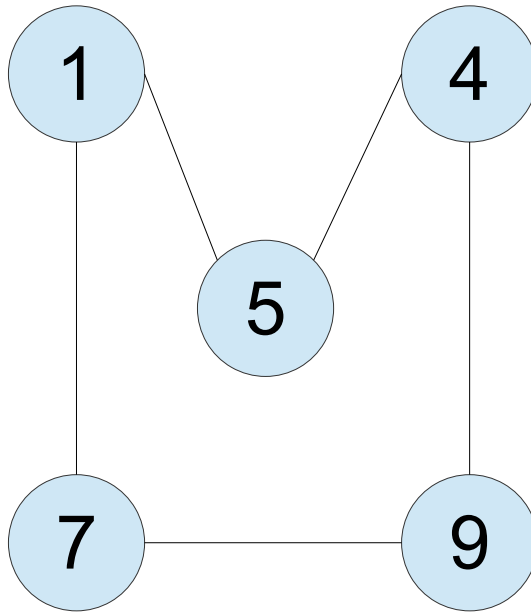


$V = ?$

$E = ?$

Grafos

- Ejemplo:

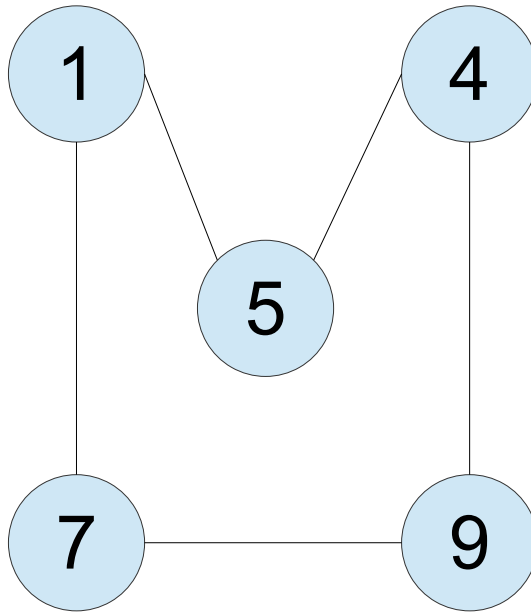


$$V = \{ 1, 4, 5, 7, 9 \}$$

$$E = \{ (1,5), (5,1), (7,1), (1,7), (7,9), (9,7), (4,5), (5,4), (4,9), (9,4) \}$$

Grafos

- Ejemplo:

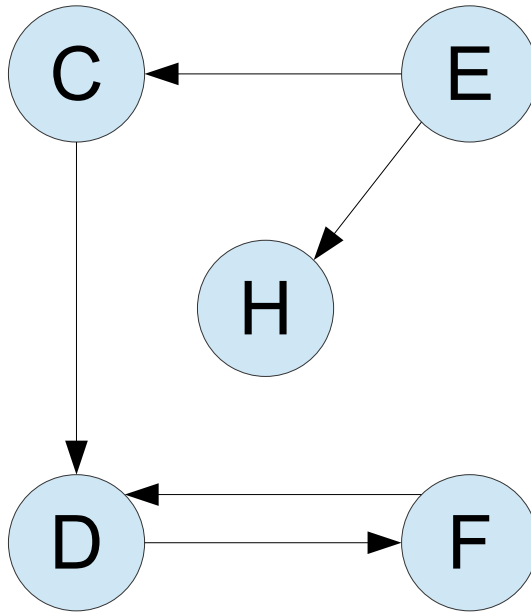


$$V = \{ 1, 4, 5, 7, 9 \}$$

$$E = \{ (1,5), (7,1), (7,9), (4,5), (4,9) \}$$

Grafos

- Ejemplo:

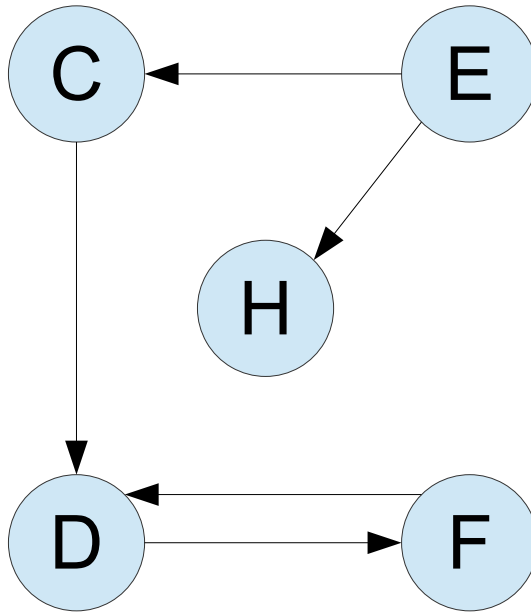


$V = ?$

$E = ?$

Grafos

- Ejemplo:



$$V = \{ C, D, E, F, H \}$$

$$E = \{ (C,D), (D,F), (F,D), (E,C), (E,H) \}$$

Tipos de grafos

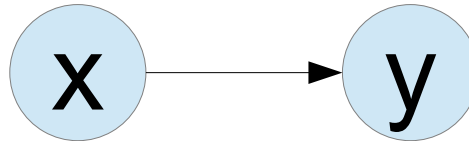
- Grafos dirigidos (digrafos):



- y es adyacente a $x \rightarrow (x,y) \in E$.
 x no es adyacente a y .
- x es el predecesor de y .
- y es el sucesor de x .
- x es el origen (fuente) de la arista.
 y es el destino (objetivo) de la arista.

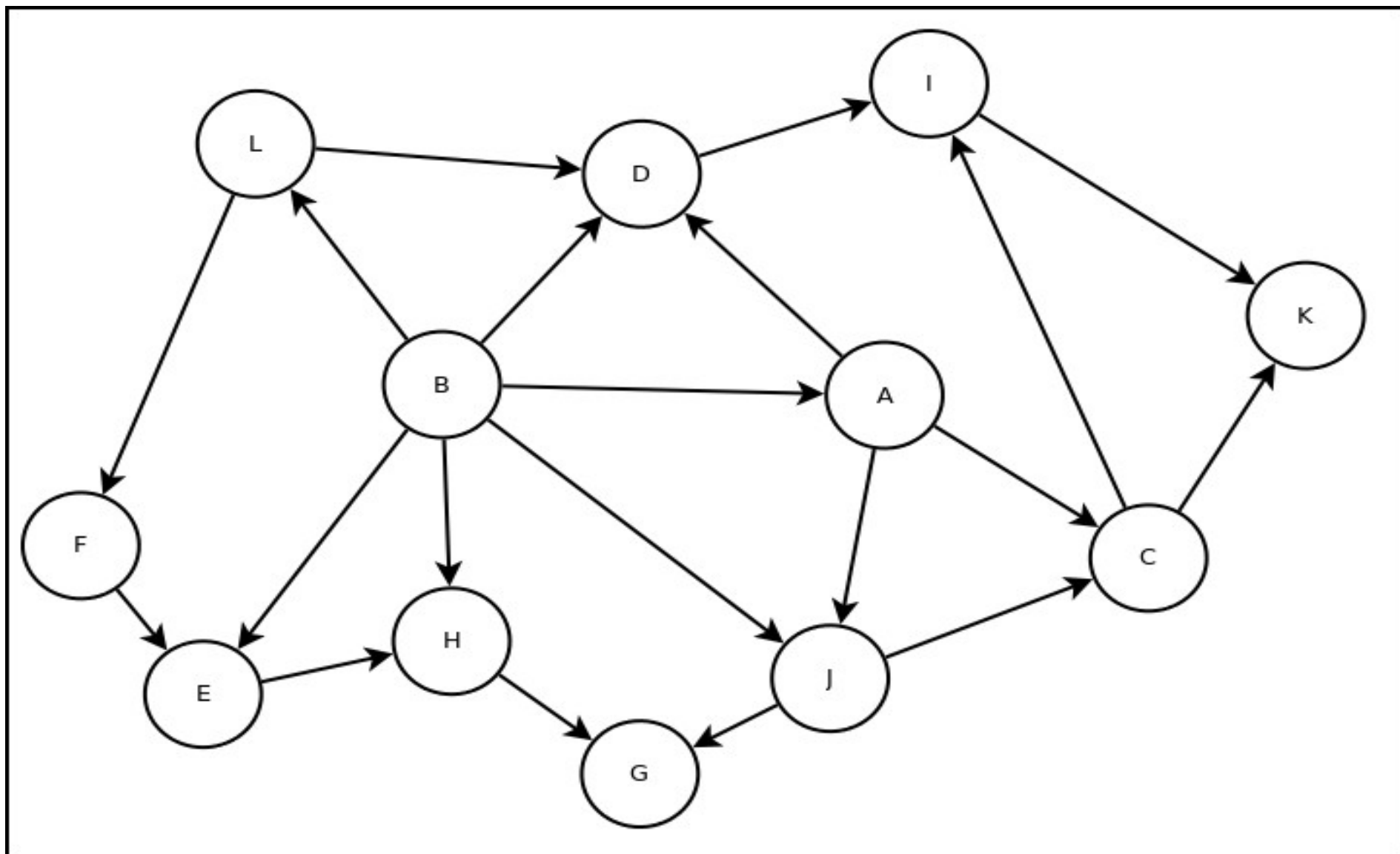
Tipos de grafos

- Grafos dirigidos (digrafos):



- Grado u orden del nodo: # aristas conectadas.
- Grado u orden de entrada: # aristas donde el nodo es el destino.
- Grado u orden de salida: # aristas donde el nodo es el origen.
- Nodo fuente: orden de entrada es cero.
- Nodo sumidero: orden de salida es cero.

Grafo dirigido



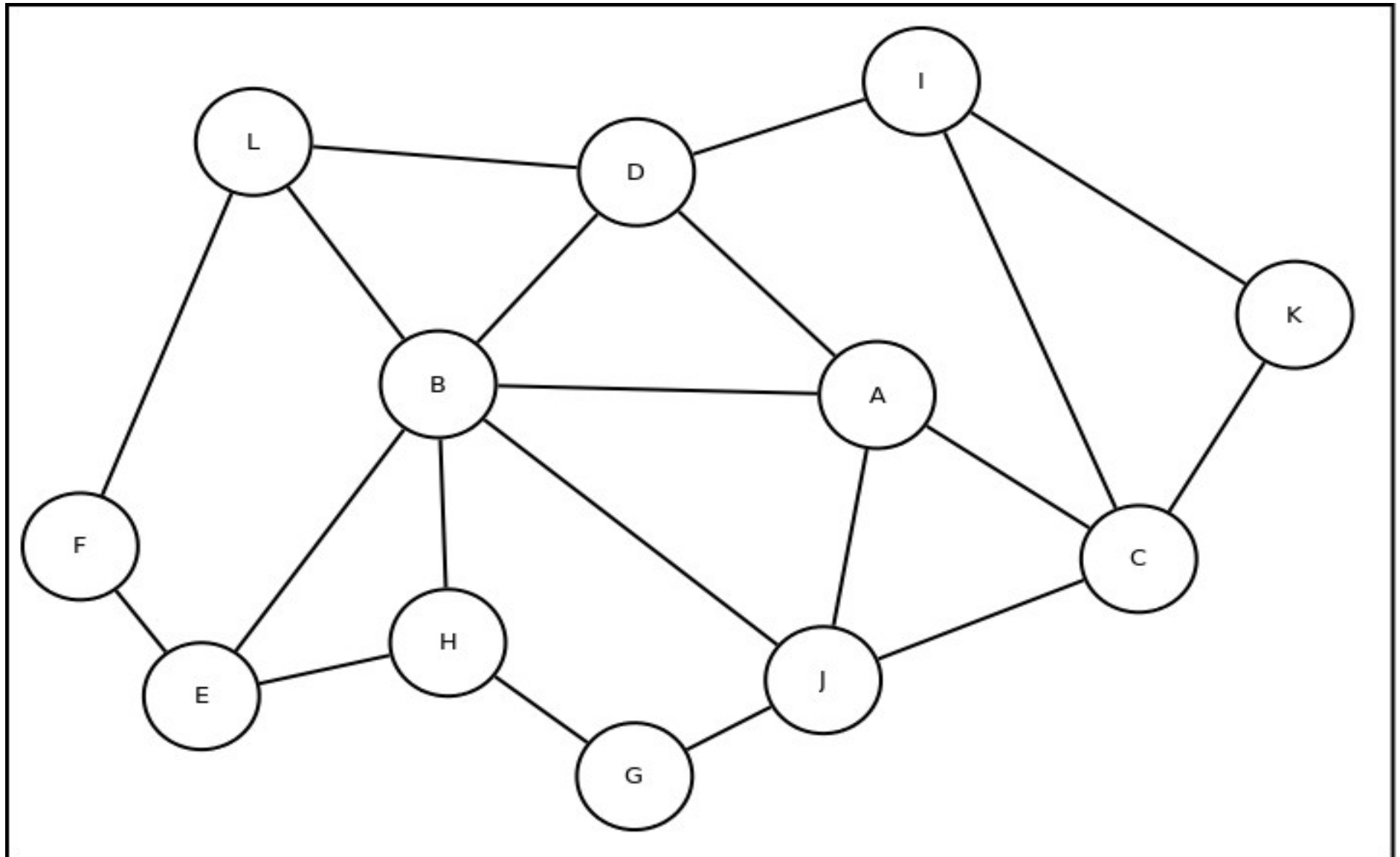
Tipos de grafos

- Grafos no dirigidos:



- x es adyacente a y, y es adyacente a x
 $\rightarrow (x,y) \in E$.
- x, y son considerados vecinos.
- Grado u orden del nodo: # aristas conectadas.

Grafo no dirigido



Grafos

- Ruta o camino:

secuencia de vértices w_1, w_2, \dots, w_n tales que $(w_i, w_{i+1}) \in E$, $1 \leq i < n$, y cada vértice es único, a excepción del primero (que puede ser también el último).

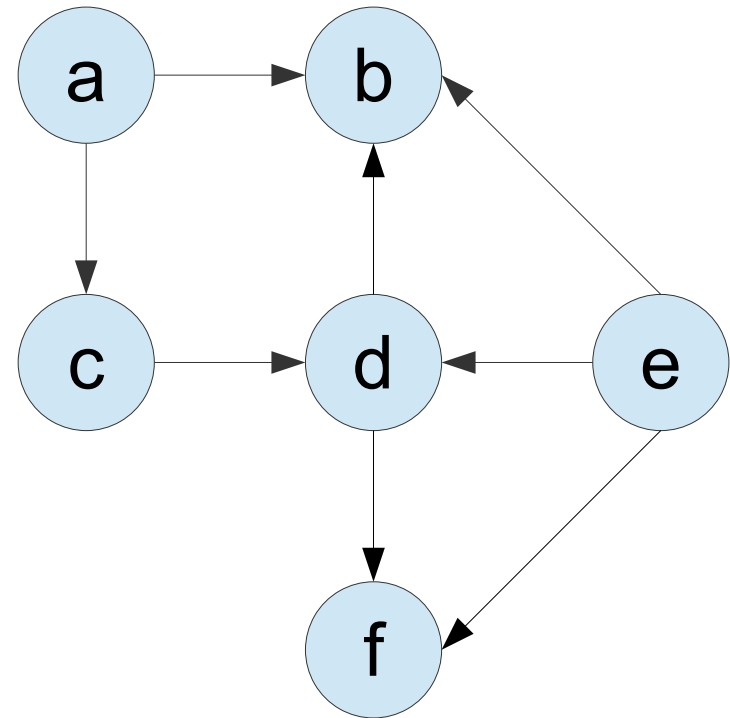
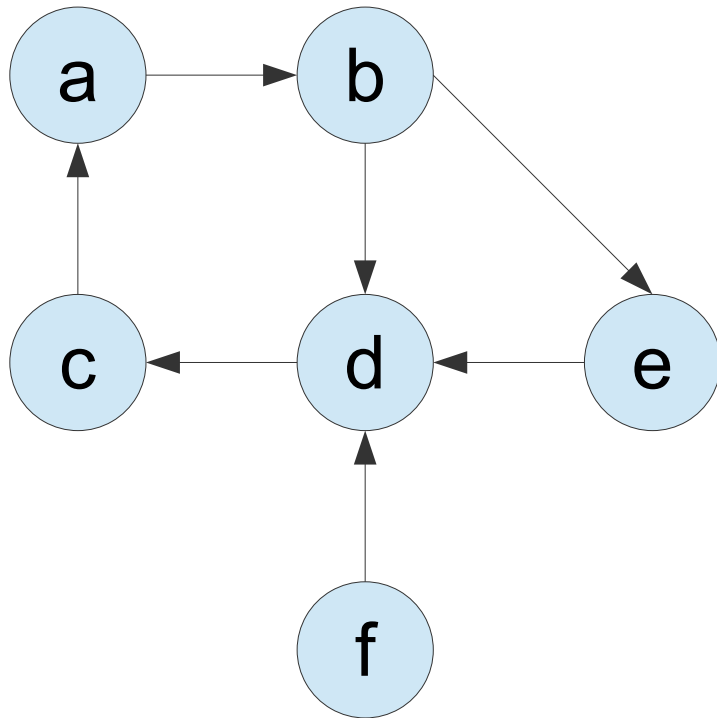
- Longitud del camino: cantidad de aristas.

Grafo

- Camino acíclico: ruta donde cada vértice es único.
- Camino cíclico: tiene al menos dos vértices, y $w_1 = w_n$ (inicia y termina en el mismo vértice).
- Una arista con el mismo vértice de origen y destino es conocida como un ciclo (*loop*).

Grafos

- ¿Cíclicos o acíclicos?



Grafos

- Un grafo dirigido sin caminos cíclicos se conoce como Grafo Dirigido Acíclico (GDA).
- Propiedades matemáticas muy importantes: ordenamiento topológico.

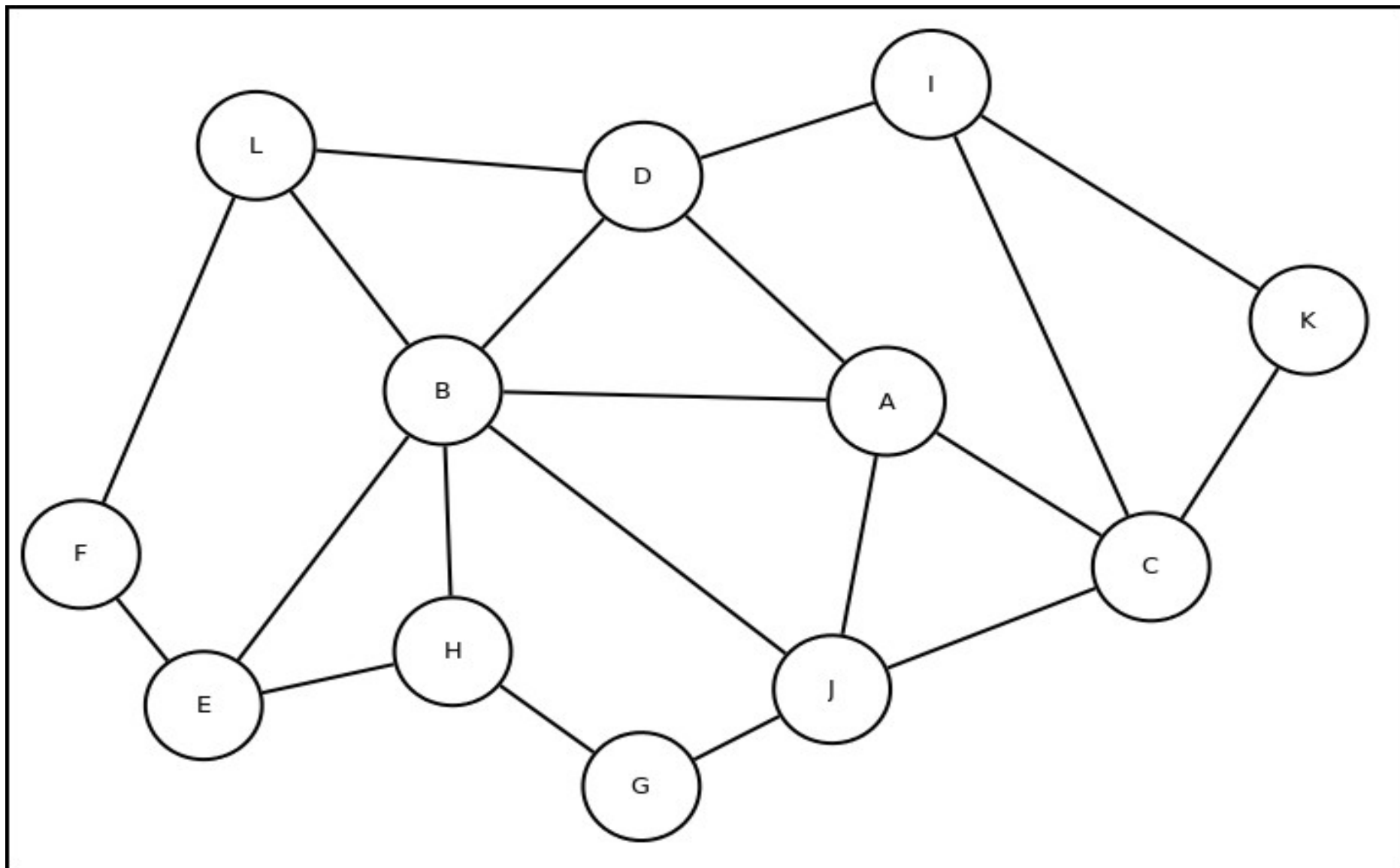
Ordenamiento de los vértices de manera tal que para cada arista, el origen siempre se encuentra antes que el destino en el ordenamiento.

- Aplicaciones:
 - Algoritmos de recorrido.
 - Redes de procesamiento de elementos.
 - Árboles.

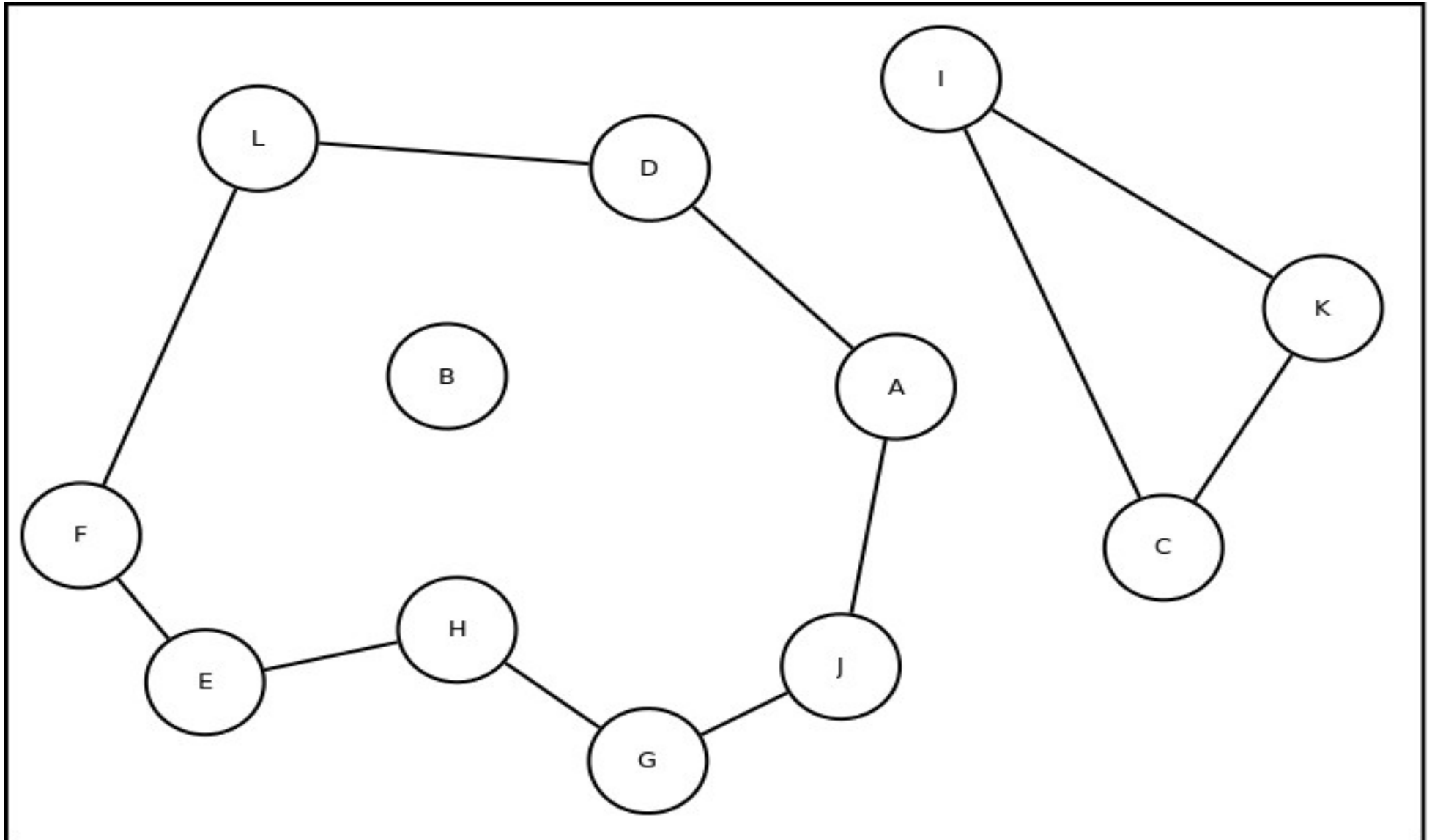
Grafos

- Un grafo no dirigido está conectado (o es un grafo conexo) si existe un camino desde cada vértice a cada uno de los otros vértices.
- Un grafo dirigido está fuertemente conectado (o es fuertemente conexo) si existe un camino desde cada vértice a cada uno de los otros vértices.
- Un grafo dirigido está débilmente conectado (o es débilmente conexo) si existe un camino desde cada vértice a cada uno de los otros vértices, sin importar la dirección de la arista.

Grafo conexo

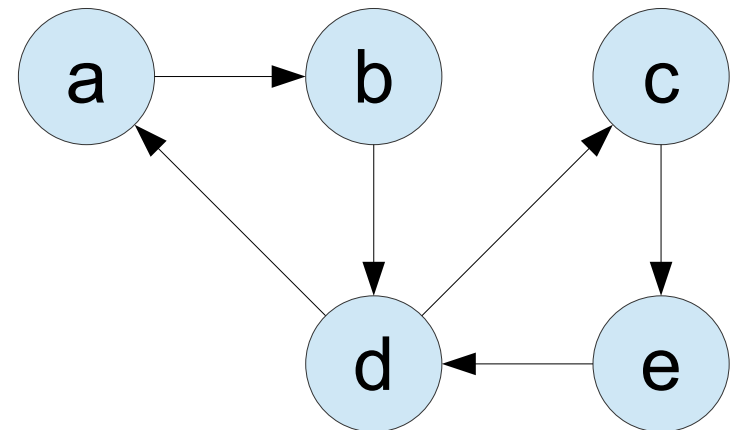
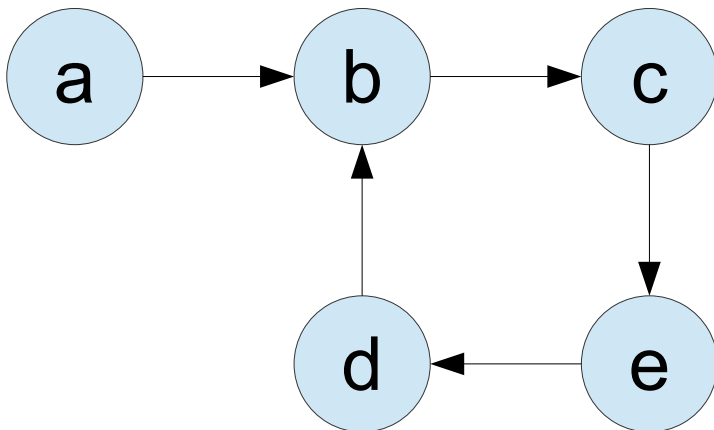
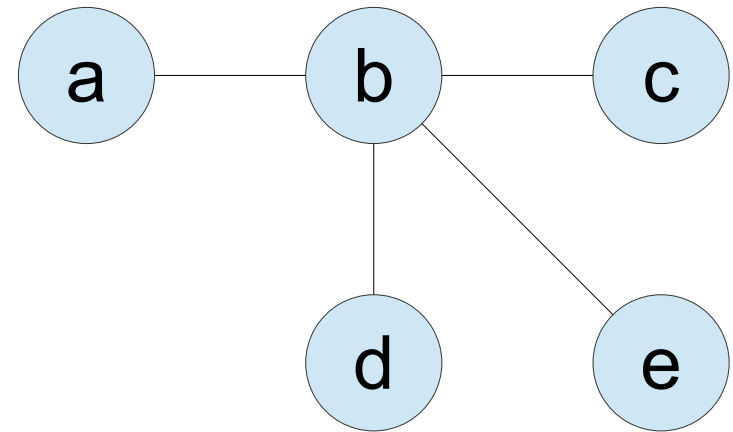
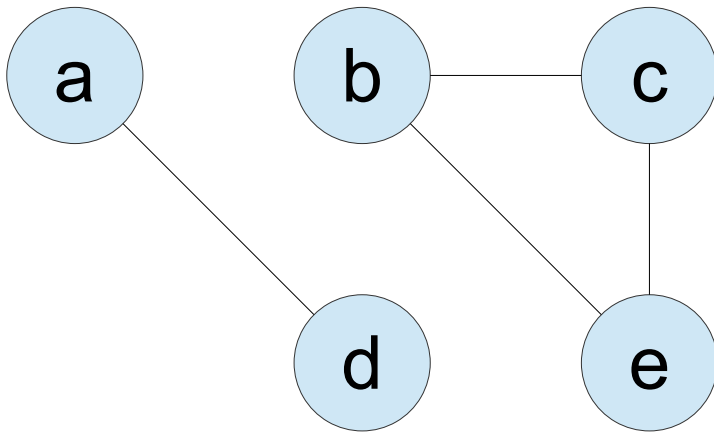


Grafo desconexo



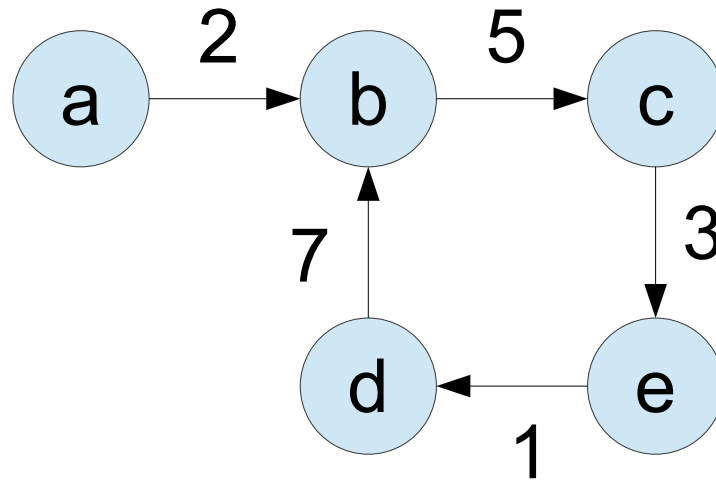
Grafos

- ¿Conectados, fuertemente conectados o débilmente conectados?



Grafos

- Un grafo es conocido como grafo etiquetado o grafo con pesos si cada arista tiene asociado un peso o valor.



Grafos

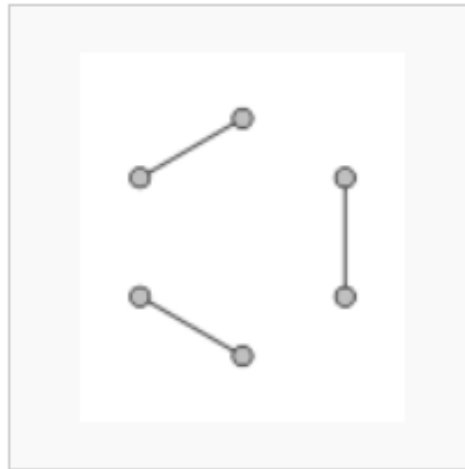
- Un grafo sin vértices ni aristas es llamado el grafo nulo o grafo vacío.
- Un grafo con un solo vértice y sin aristas es conocido como un grafo trivial.

Grafos

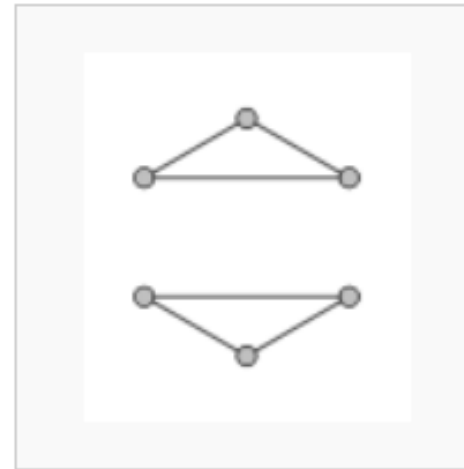
- Grafo regular: donde cada vértice tiene exactamente el mismo número de vecinos.



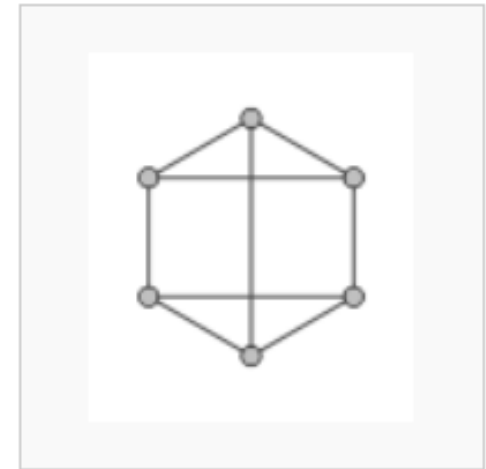
0-regular graph



1-regular graph





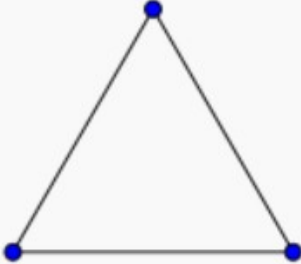
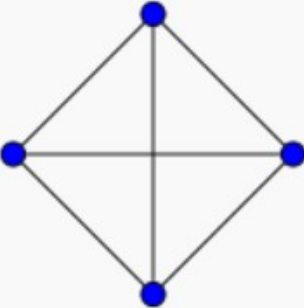
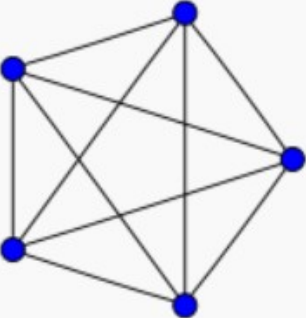
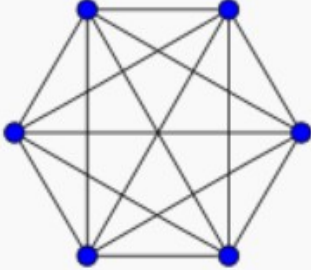
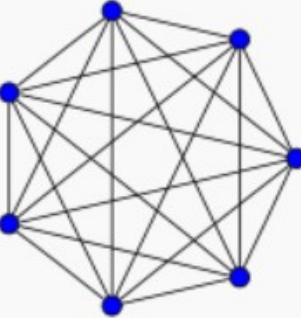
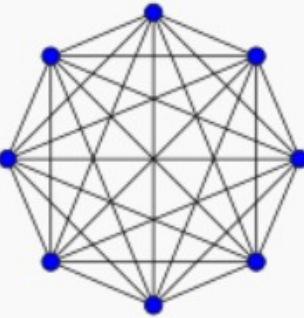
2-regular graph



3-regular graph

Grafos

- Grafo completo: donde cada par de vértices tiene una arista que los conecta.

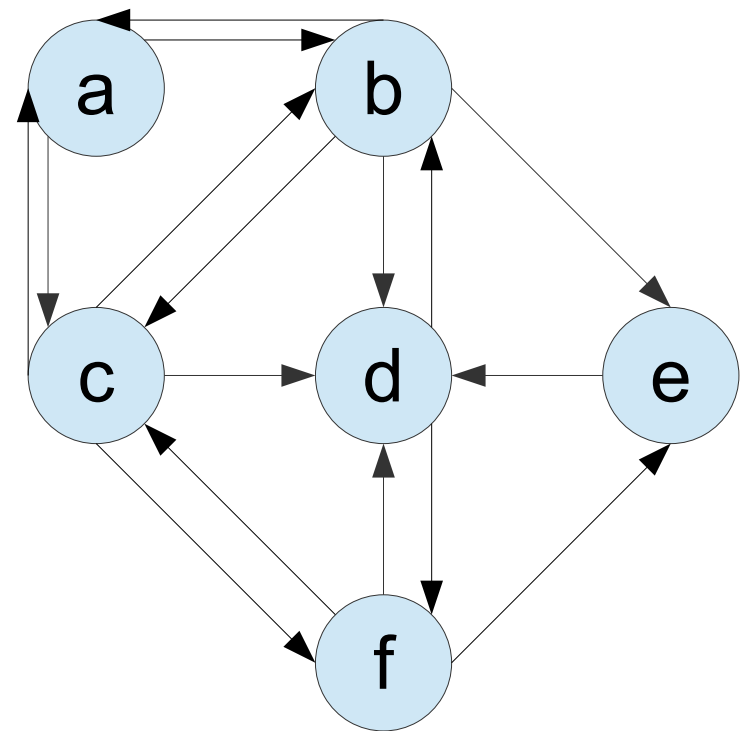
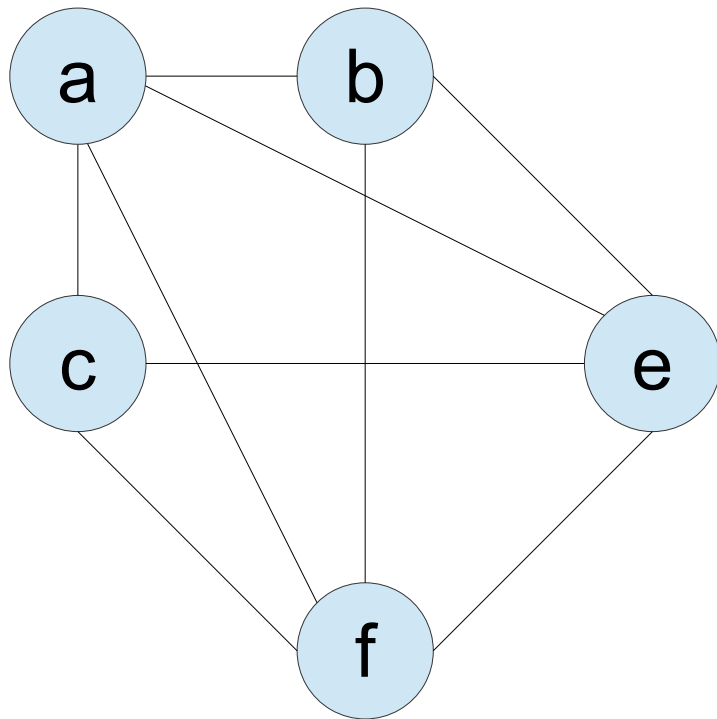
$K_1: 0$	$K_2: 1$	$K_3: 3$	$K_4: 6$
			
$K_5: 10$	$K_6: 15$	$K_7: 21$	$K_8: 28$
			
$K_9: 36$	$K_{10}: 45$	$K_{11}: 55$	$K_{12}: 66$

Grafos

- Grafo completo: donde cada par de vértices tiene una arista que los conecta.
 - Un grafo dirigido también puede ser un grafo completo, en ese caso, debe existir una arista desde cada vértice hacia cada uno de los otros vértices.

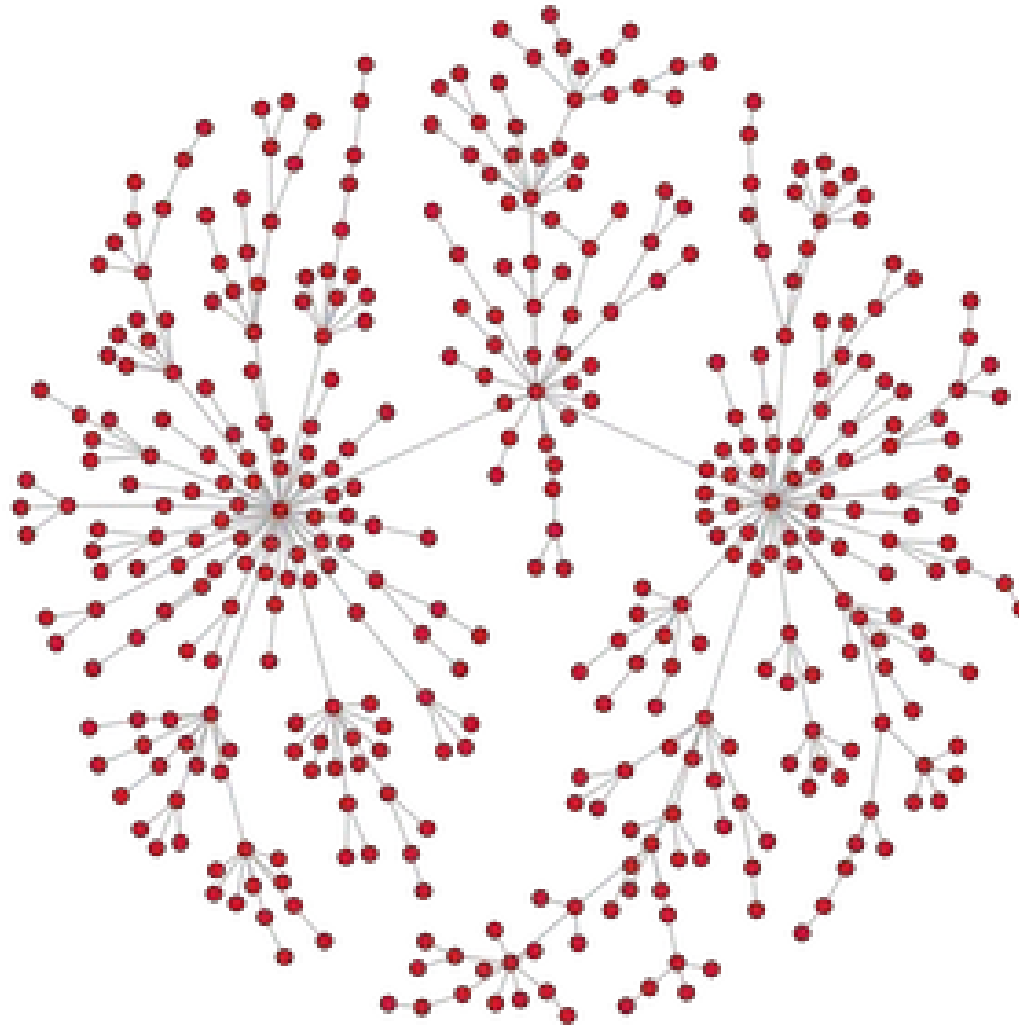
Grafos

- ¿Completos o no completos?



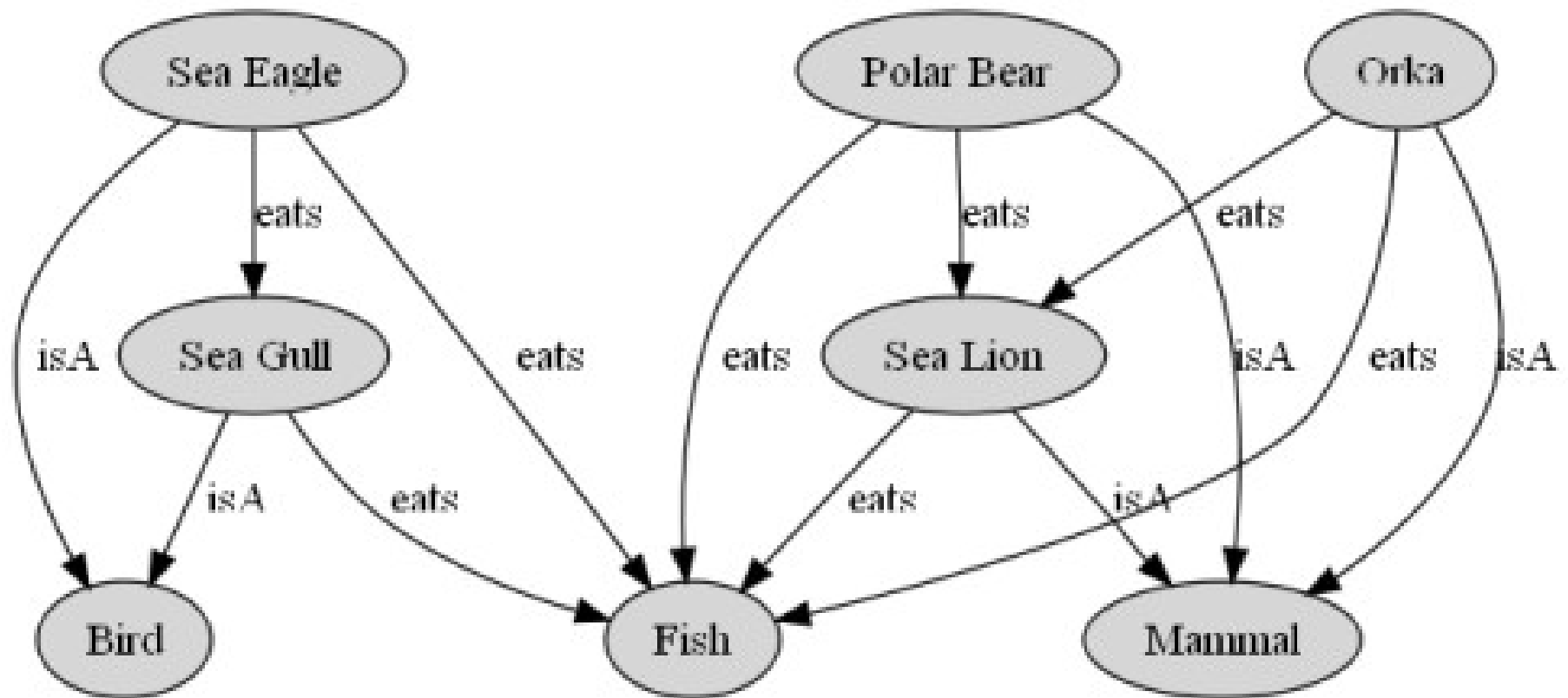
Grafos

- Ejemplos: relaciones entre personas.



Grafos

- Ejemplos: relaciones entre conceptos.



Grafos

- Ejemplos: relaciones geográficas.



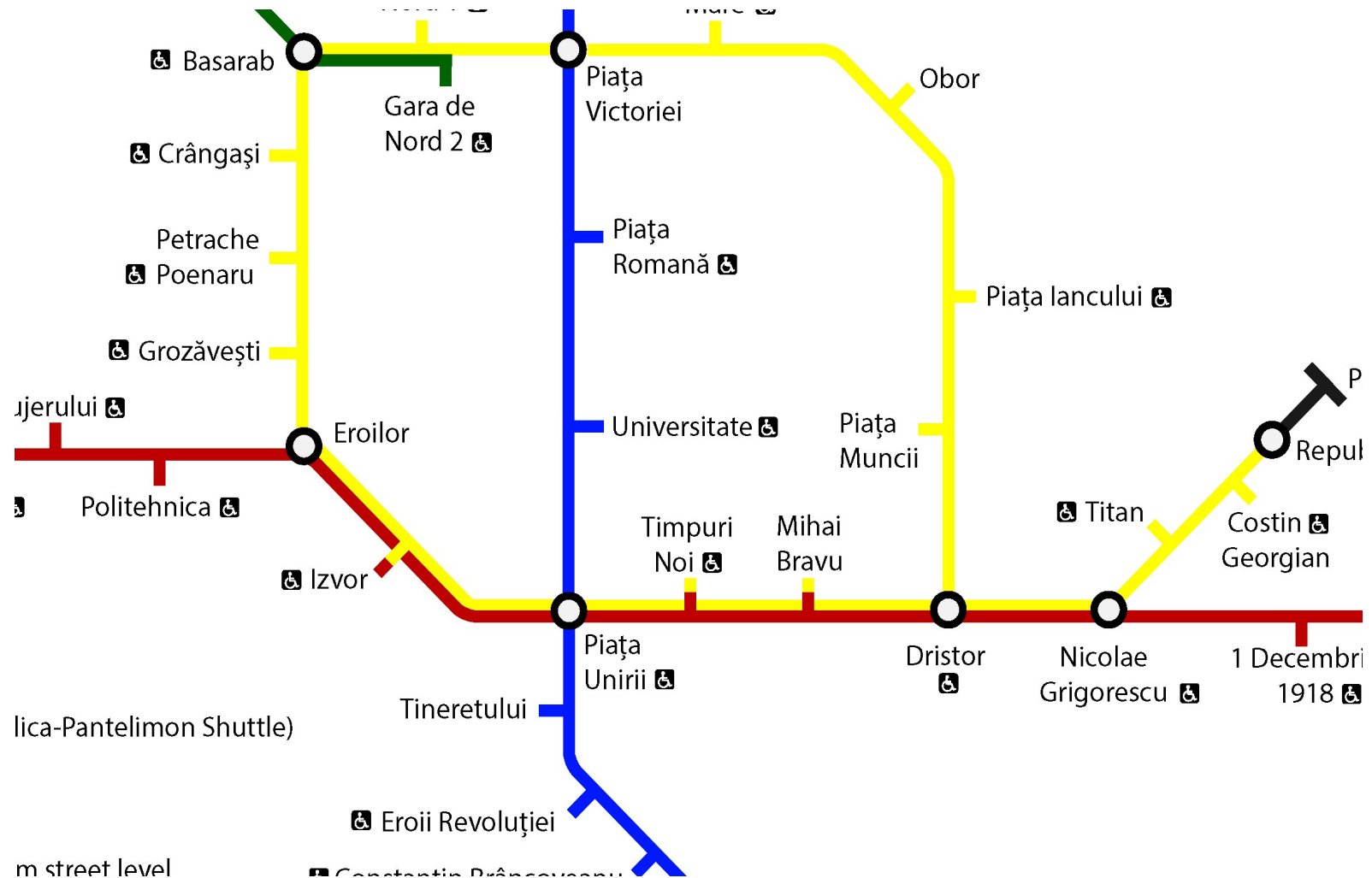
Grafos

- Ejemplos: relaciones geográficas.



Grafos

- Ejemplos: representación de redes.

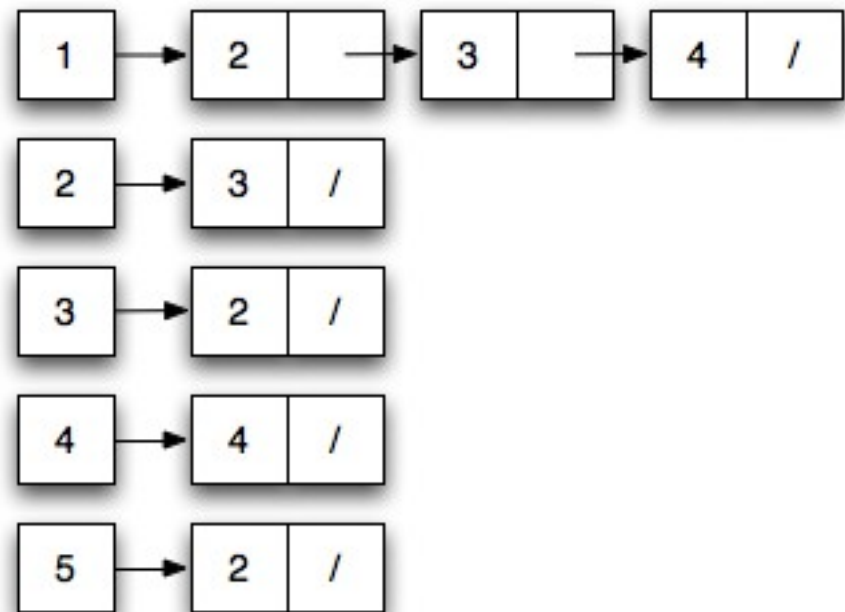
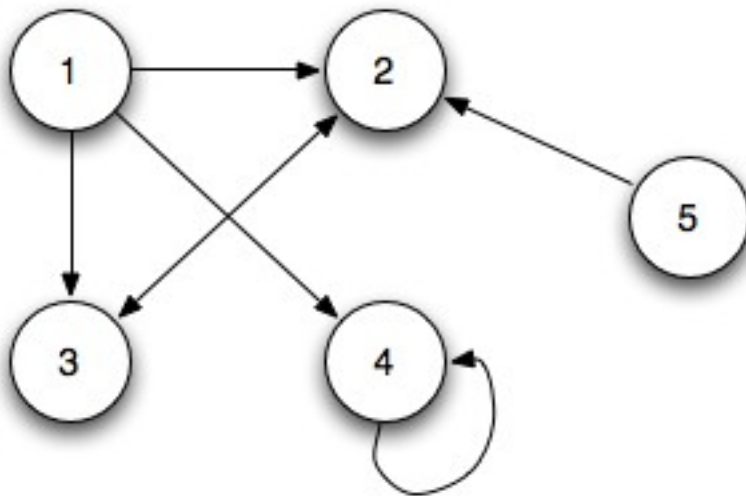


Implementación de Grafos

- Lista de adyacencia.
- Lista de incidencia.
- Matriz de adyacencia.
- Matriz de incidencia.

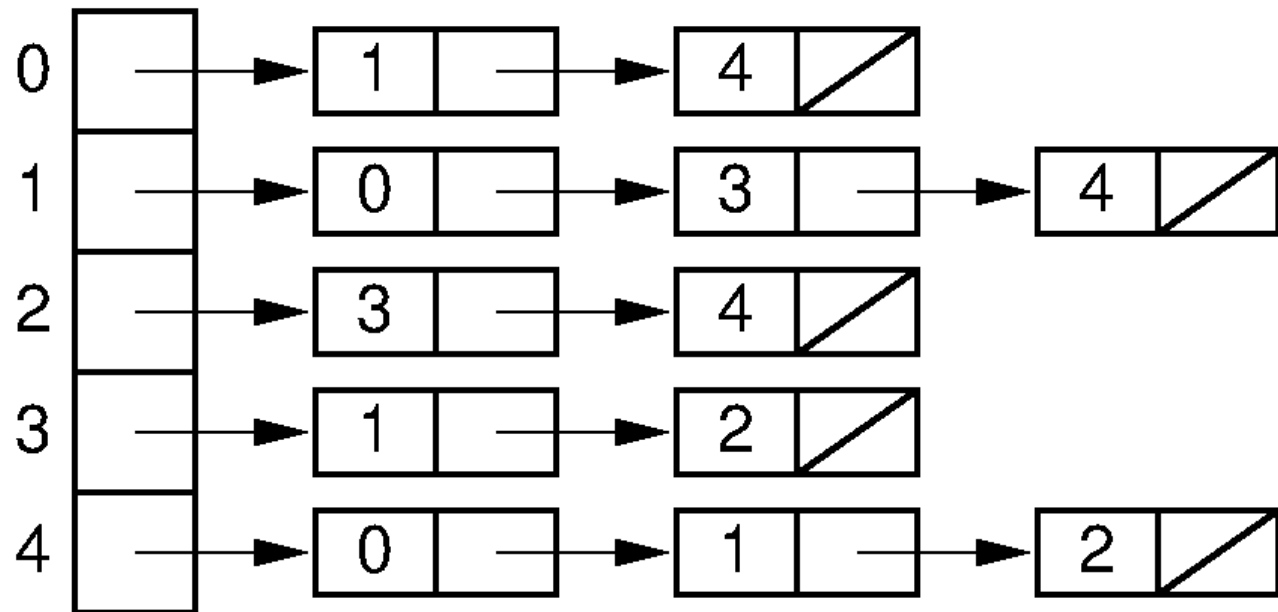
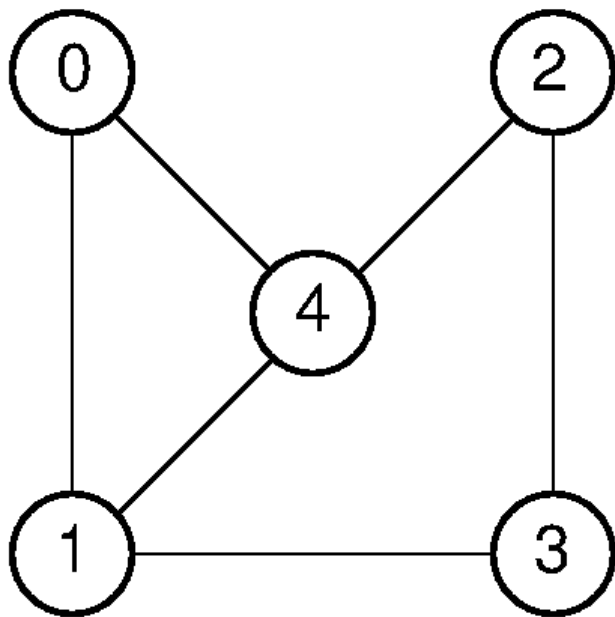
Implementación de Grafos

- Lista de adyacencia:
cada vértice almacena una lista de adyacencia.



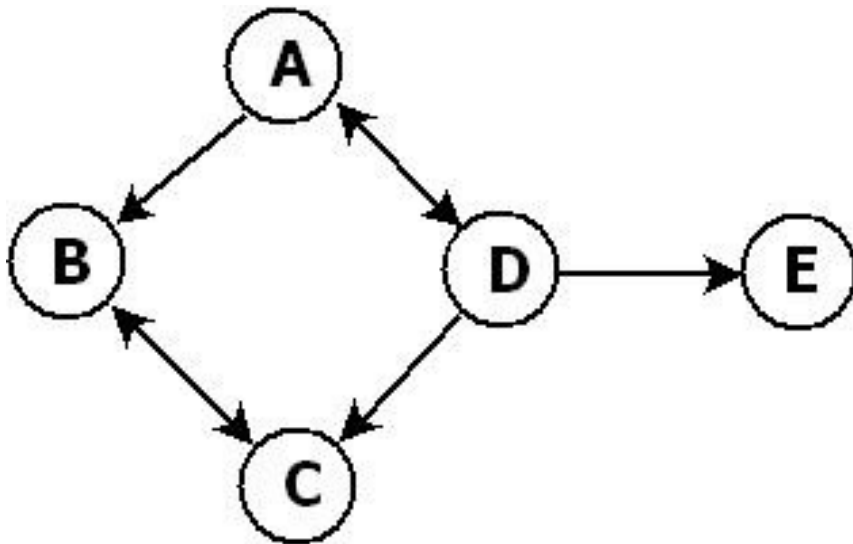
Implementación de Grafos

- Lista de adyacencia:
cada vértice almacena una lista de adyacencia.



Implementación de Grafos

- Lista de incidencia:
cada vértice almacena las aristas incidentes, y/o
cada arista almacena los vértices incidentes.



A: {<A, B>, <A, D>}

B: {<B, C>}

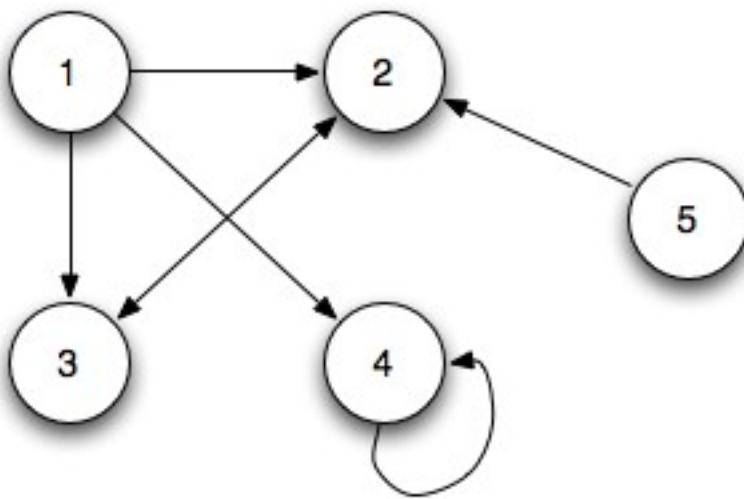
C: {<C, B>}

D: {<D, A>, <D, C>, <D, E>}

E: {}

Implementación de Grafos

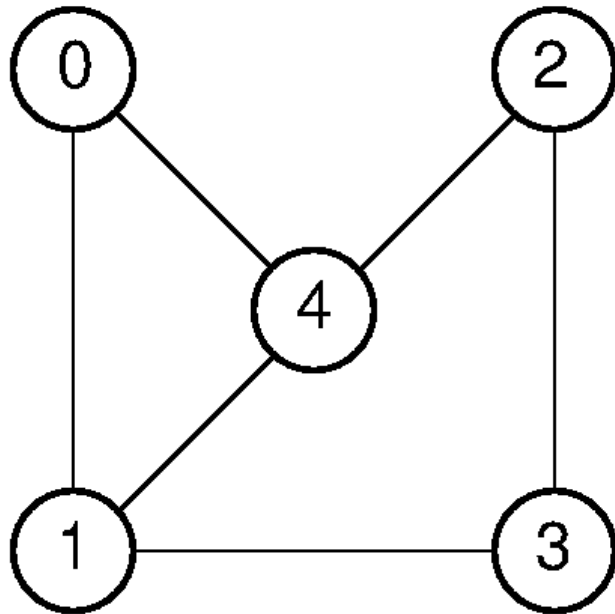
- Matriz de adyacencia:
matriz 2D, vértices origen en las filas y vértices destino en las columnas (costo de arista).



	1	2	3	4	5
1	0	1	1	1	0
2	0	0	1	0	0
3	0	1	0	0	0
4	0	0	0	1	0
5	0	1	0	0	0

Implementación de Grafos

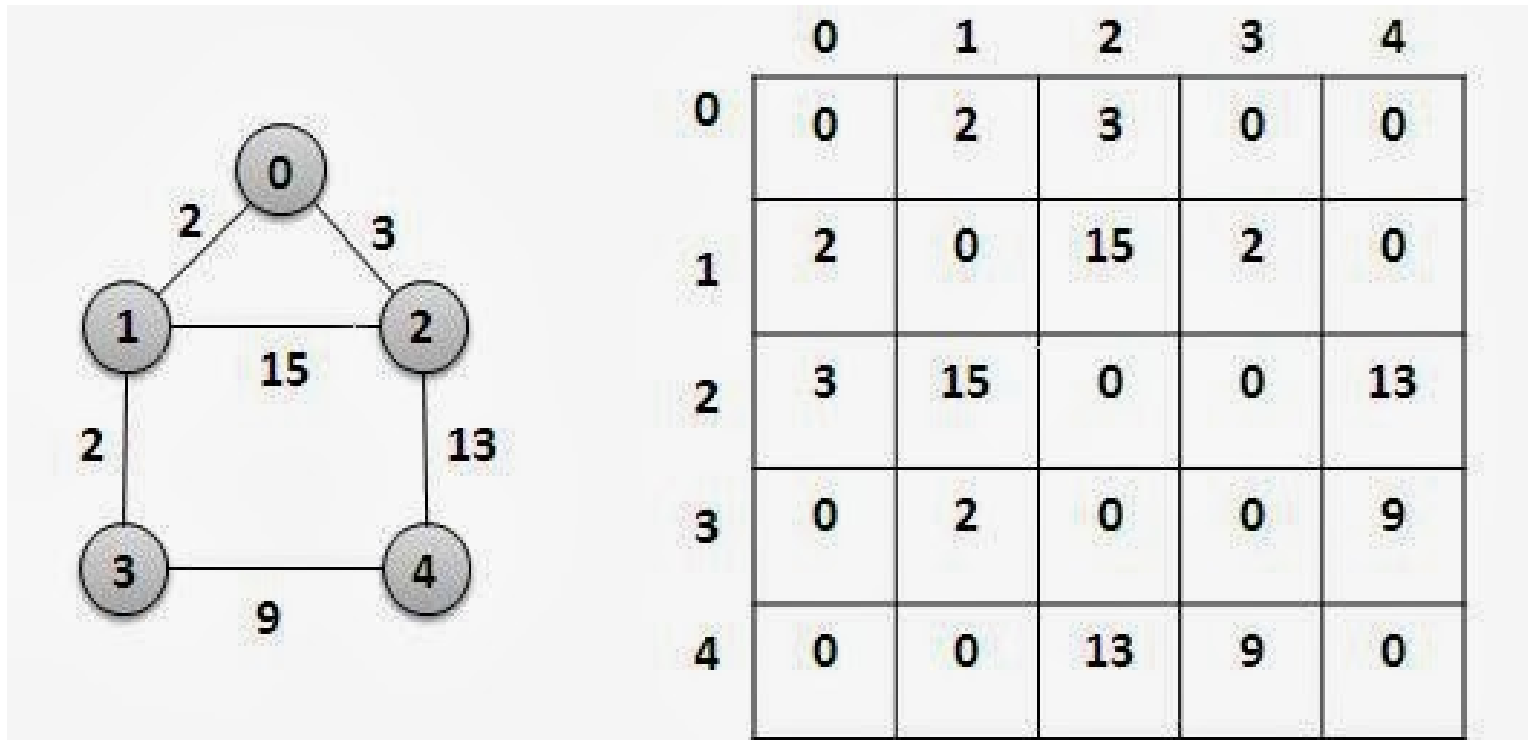
- Matriz de adyacencia:
matriz 2D, vértices origen en las filas y vértices destino en las columnas (costo de arista).



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

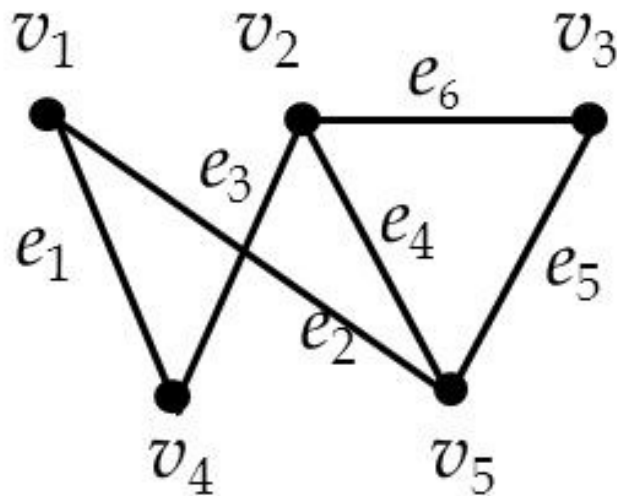
Implementación de Grafos

- Matriz de adyacencia:
matriz 2D, vértices origen en las filas y vértices destino en las columnas (costo de arista).



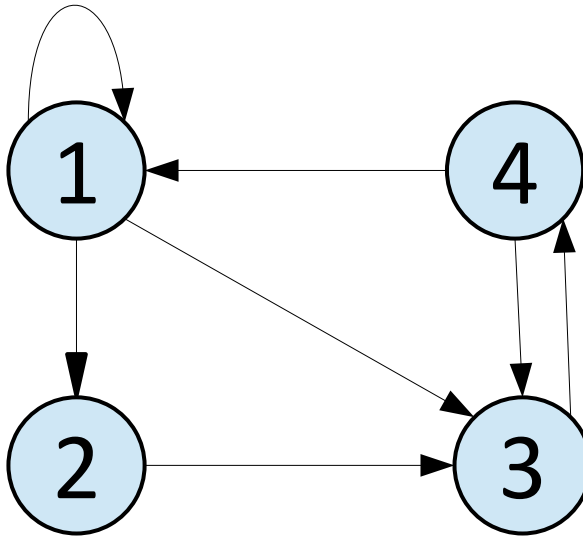
Implementación de Grafos

- Matriz de incidencia:
matriz booleana 2D, filas representan vértices,
columnas representan aristas (incidencia).



	e_1	e_2	e_3	e_4	e_5	e_6
v_1	1	1	0	0	0	0
v_2	0	0	1	1	0	1
v_3	0	0	0	0	1	1
v_4	1	0	1	0	0	0
v_5	0	1	0	1	1	0

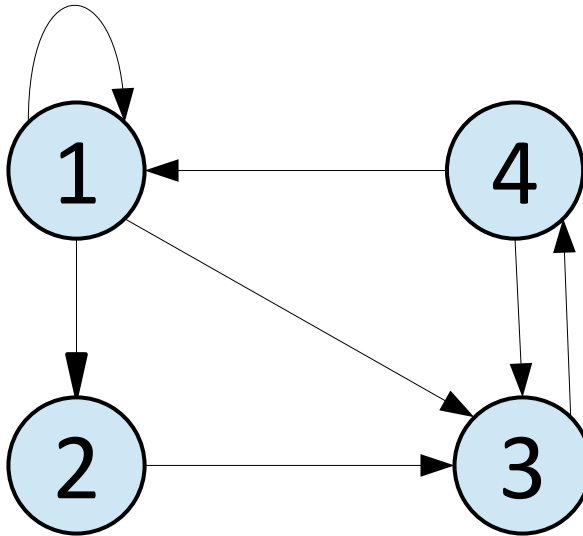
Implementación de Grafos



Adjacency matrix

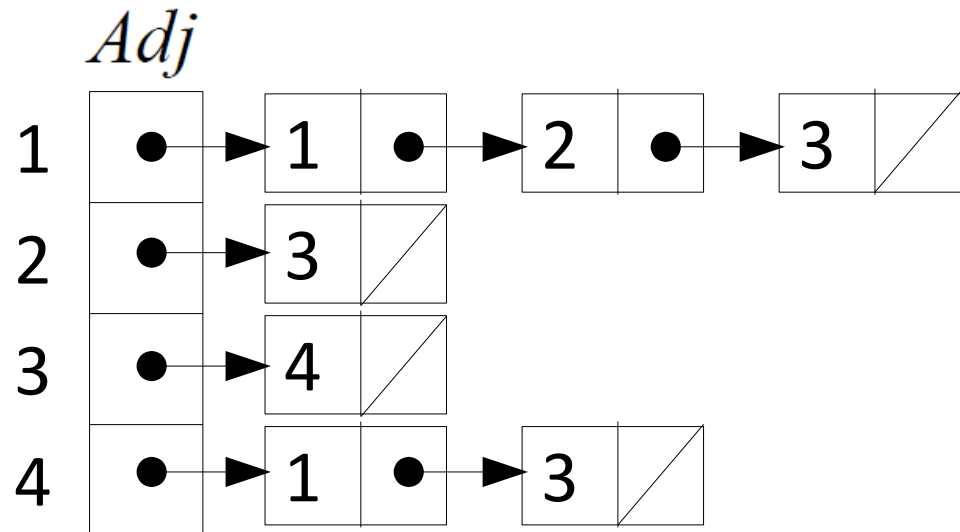
Adjacency list

Implementación de Grafos



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

Adjacency matrix



Adjacency list

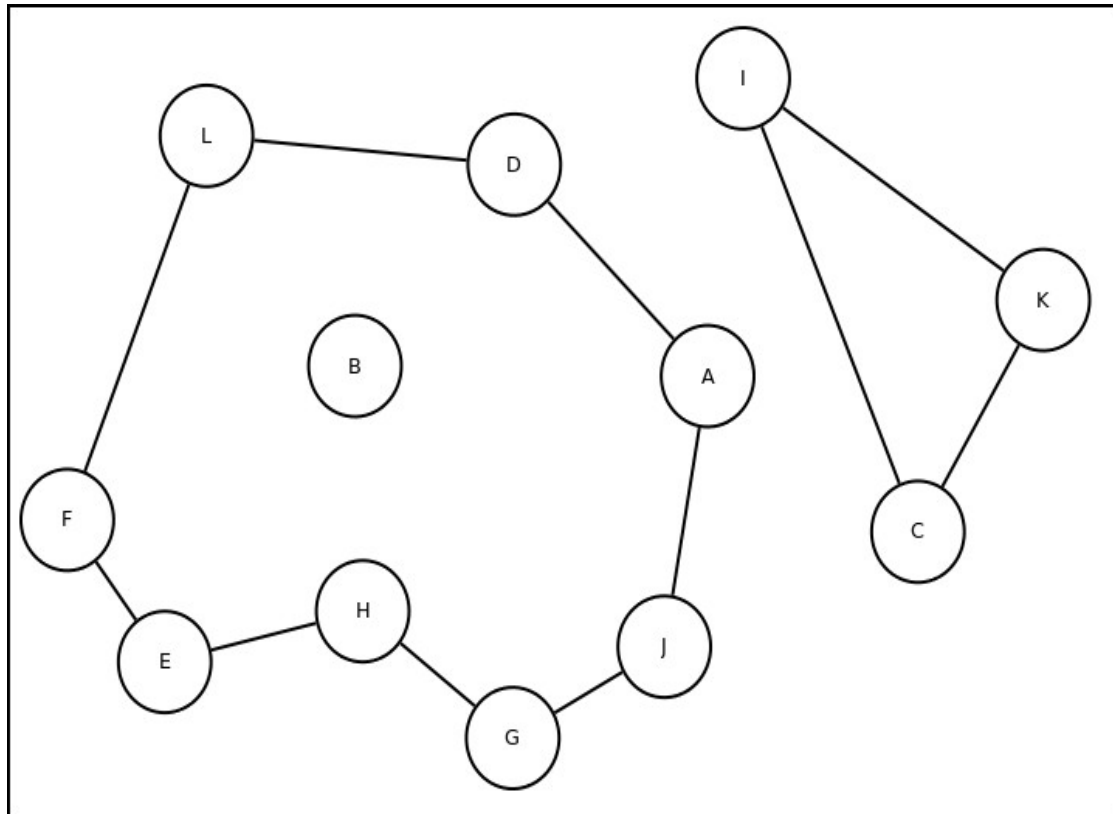
Implementación de Grafos

{A,B,C,D,E,F,G,H,I,J,K,L}

	A	B	C	D	E	F	G	H	I	J	K	L
A				1						1		
B												
C									1		1	
D	1											1
E						1		1				
F					1							1
G								1		1		
H					1		1					
I			1								1	
J	1						1					
K			1						1			
L				1		1						

Implementación de Grafos

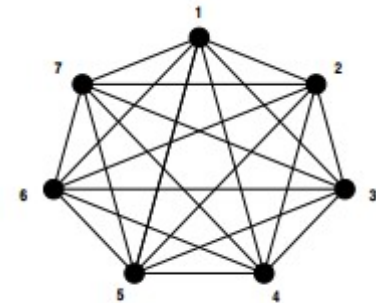
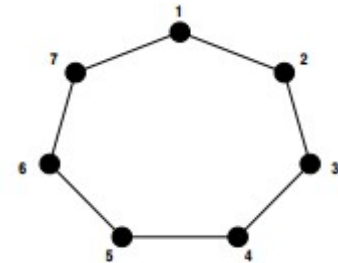
{A,B,C,D,E,F,G,H,I,J,K,L}



	A	B	C	D	E	F	G	H	I	J	K	L
A				1						1		
B												
C									1		1	
D	1											1
E						1		1				
F					1							1
G								1		1		
H					1		1					
I			1								1	
J	1						1					
K			1						1			
L				1	1							

Recorridos en Grafos

- Plano
- Preorden
- Niveles / Vecindario
- Eulerianos
 - Todas las aristas una vez
- Hamilton
 - Todos los vértices una vez



Grafos

- Trabajo en casa:
 - Diseñar el TAD Grafo (escoger la estructura de implementación).
 - Generar diseño textual (descripción de TADs según plantilla).

Referencias

- L. Joyanes Aguilar, I. Zahonero. Algoritmos y estructuras de datos: una perspectiva en C. McGraw-Hill, 2004.
- www.cs.umd.edu/~mount/420/Lects/420lects.pdf
- people.cs.clemson.edu/~pargas/courses/cs212/common/notes/ppt/17aGraphs.ppt
- [en.wikipedia.org/wiki/Graph_\(abstract_data_type\)](http://en.wikipedia.org/wiki/Graph_(abstract_data_type))
- [en.wikipedia.org/wiki/Graph_\(mathematics\)](http://en.wikipedia.org/wiki/Graph_(mathematics))