

Estructuras de Datos.

Grado en Informática, Ingeniería del Software y Computación

ETSI Informática

Universidad de Málaga

# Orden Topológico en Digrafos

@ José E. Gallardo, Francisco Gutiérrez, Pablo López, Blas C. Ruiz

Dpto. Lenguajes y Ciencias de la Computación

Universidad de Málaga

# Orden Topológico en DiGrafos

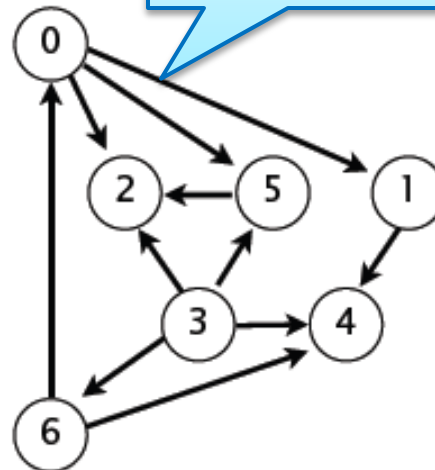
- Dado un DiGrafo acíclico (DAG, Directed Acyclic Graph), un **orden Topológico** es una relación de orden total ( $\prec$ ) entre vértices tal que:
  - Si existe un camino desde  $v$  a  $w$ , entonces  $w$  es mayor que  $v$  en el orden.  
$$v \rightarrow w \implies v \prec w$$
- Un DiGrafo puede tener diferentes Ordenes Topológicos
- Si el DiGrafo es cíclico, el Orden Topológico no existe 😞

# Orden Topológico en DiGrafos

- **Objetivo:** dado un conjunto de tareas que deben terminar antes que otras comiencen (prerrequisitos),  
¿En qué orden deben ser planificadas las tareas?
- Modelado por un DiGrafo.
  - vértice= tarea
  - Arco = prerrequisito.

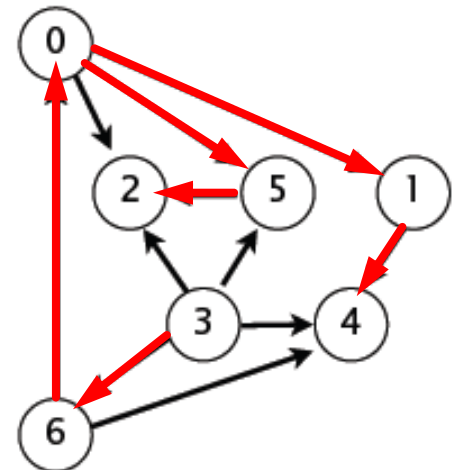
0. Algorithms  
1. Complexity Theory  
2. Artificial Intelligence  
3. Intro to CS  
4. Cryptography  
5. Scientific Computing  
6. Advanced Programming

Se debe estudiar  
*Algorithms* antes  
que *Complexity  
Theory*



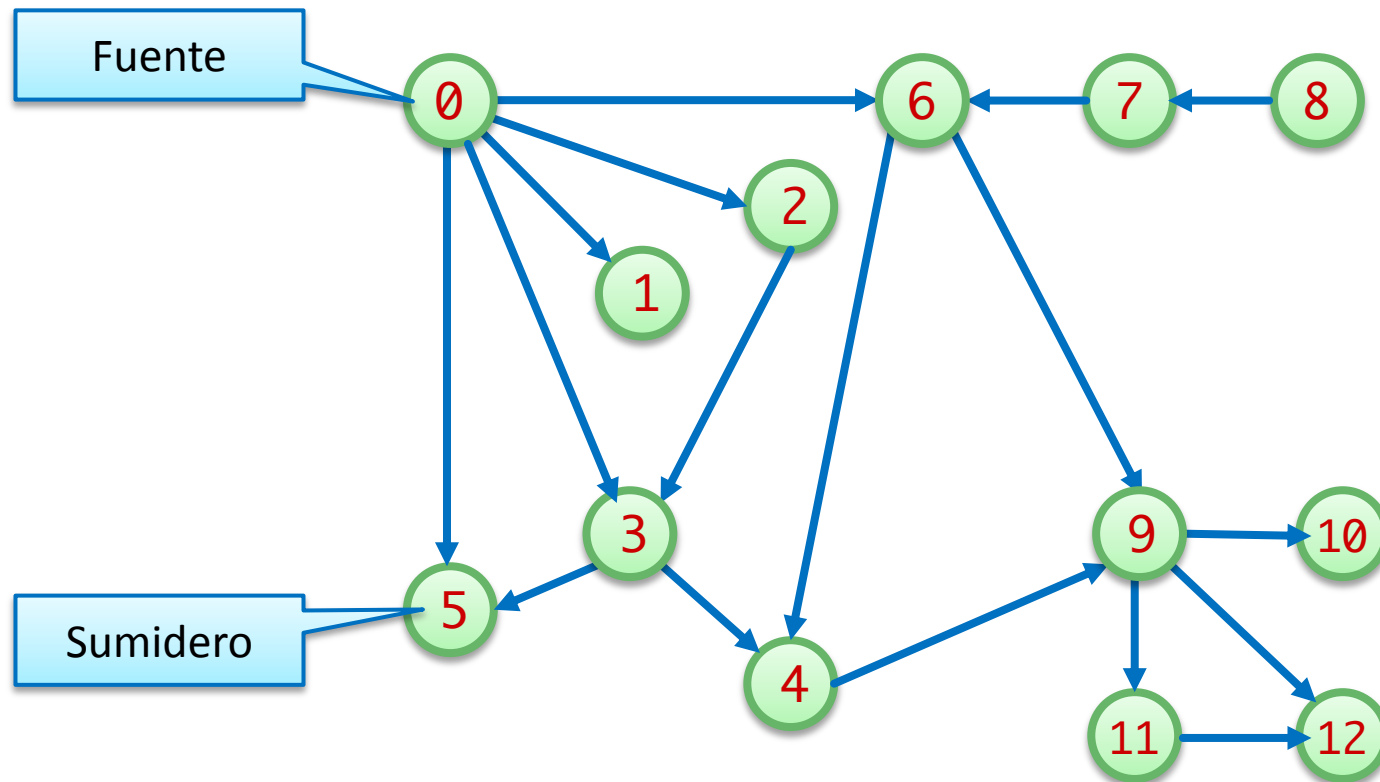
Dos Órdenes  
Topológicos

3 < 6 < 0 < 5 < 2 < 1 < 4  
3 < 6 < 0 < 1 < 4 < 5 < 2



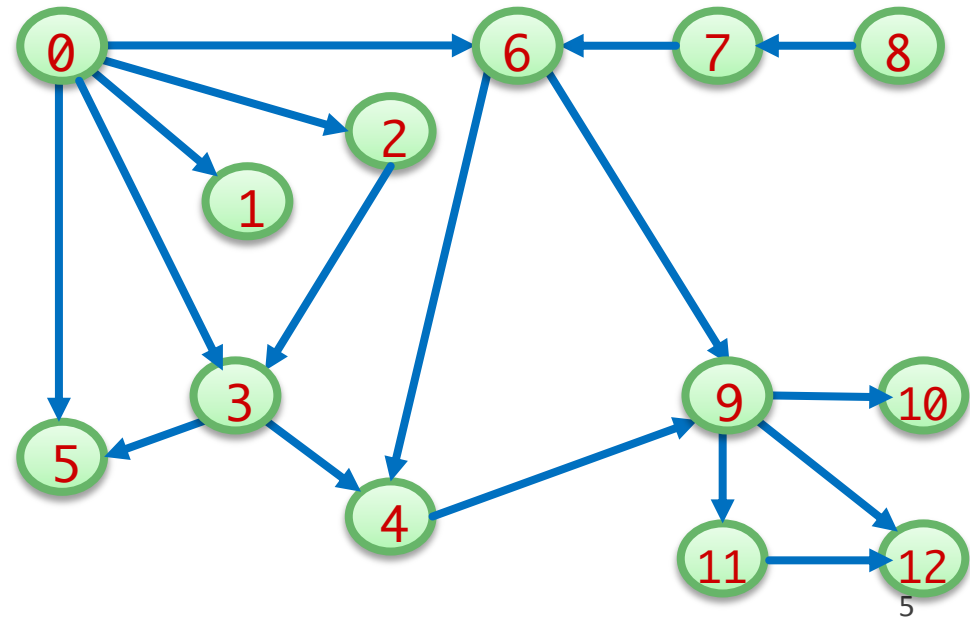
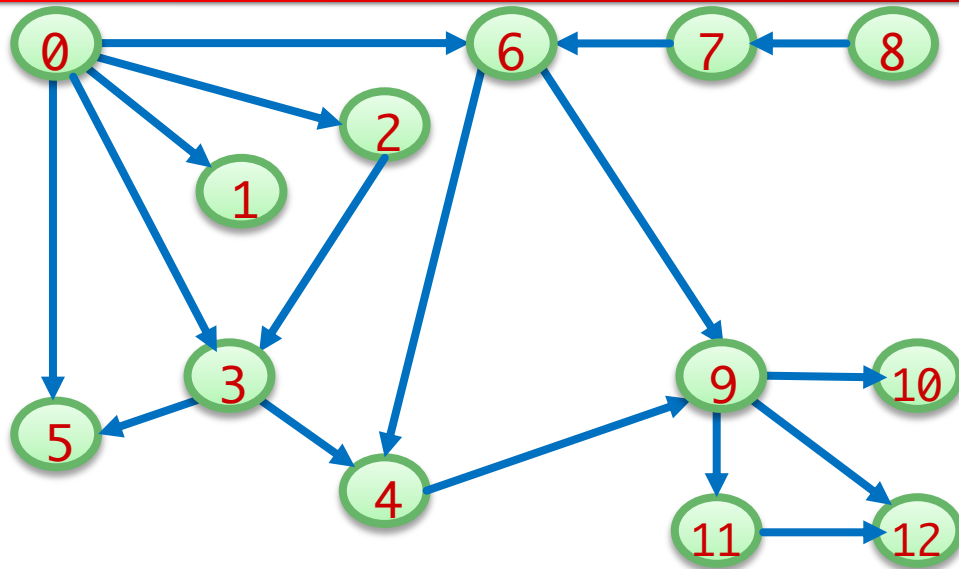
# Orden Topológico en DiGrafos

- **Fuente:** vértice cuyo grado de entrada es 0
- **Sumidero:** vértice cuyo grado de salida es 0

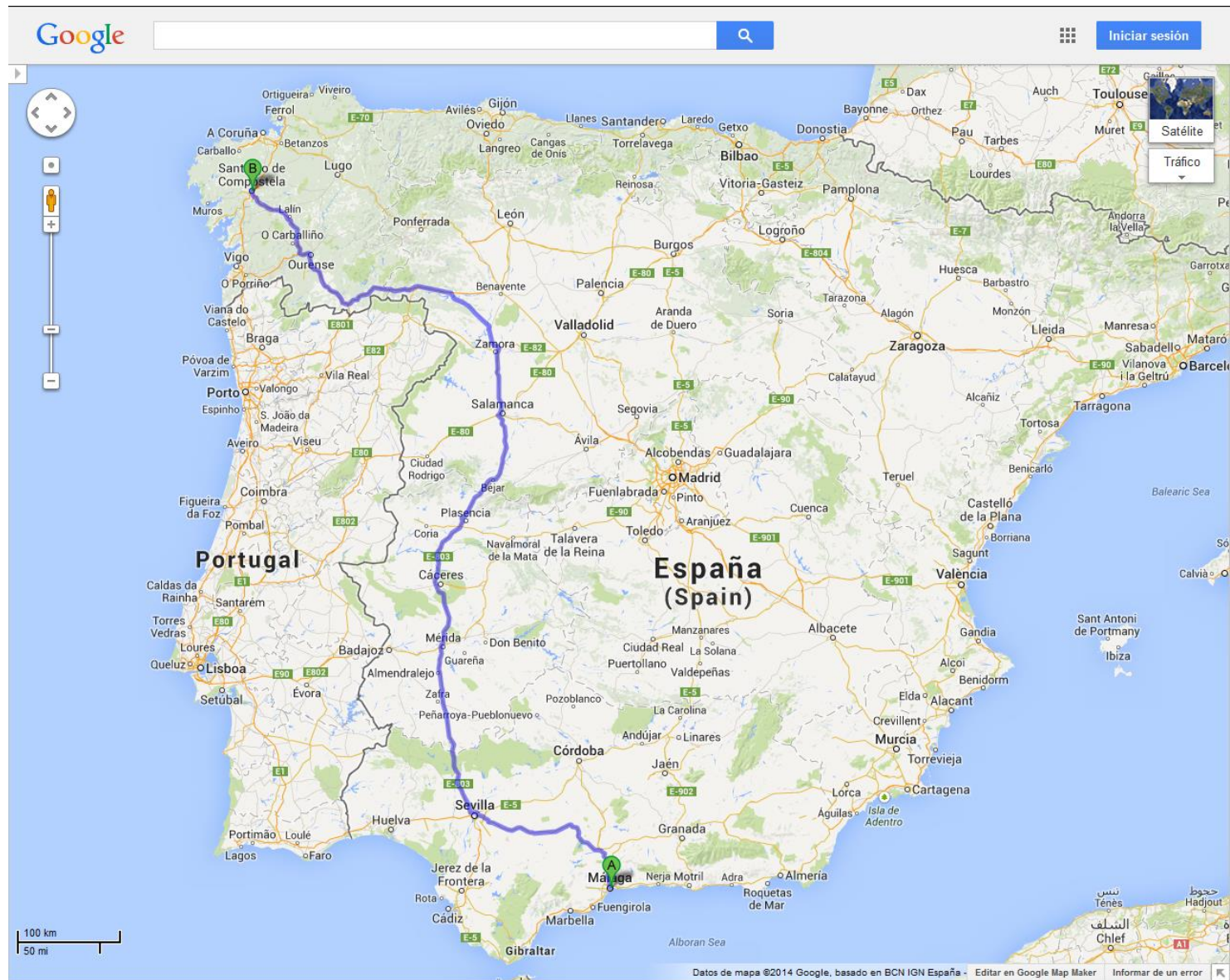


Orden Topológico: 8 < 0 < 2 < 7 < 1 < 3 < 6 < 4 < 5 < 9 < 10 < 11 < 12

# El algoritmo visto en clase clona el grafo



# Clonar no siempre es adecuado



# Un algoritmo que no clona el grafo

- Utilizaremos un diccionario:  
    vértice -> # predecesores pendientes
- El diccionario se inicializa al comenzar
- El diccionario se actualiza en cada iteración, eliminando fuentes y restando predecesores pendientes
- El grafo **no se modifica**

# Pseudocódigo

Inicializar diccionario

Mientras no hayamos acabado ni haya ciclo

    Seleccionar fuentes del diccionario

    Eliminar fuentes del diccionario

    Añadir fuentes al orden topológico

    Restar fuentes del diccionario



# Orden Topológico en DiGrafos

Inicializamos diccionario con grados de entrada

**vértice -> # predecesores pendientes**

0 -> 0

1 -> 1

2 -> 1

3 -> 2

4 -> 2

5 -> 2

6 -> 2

7 -> 1

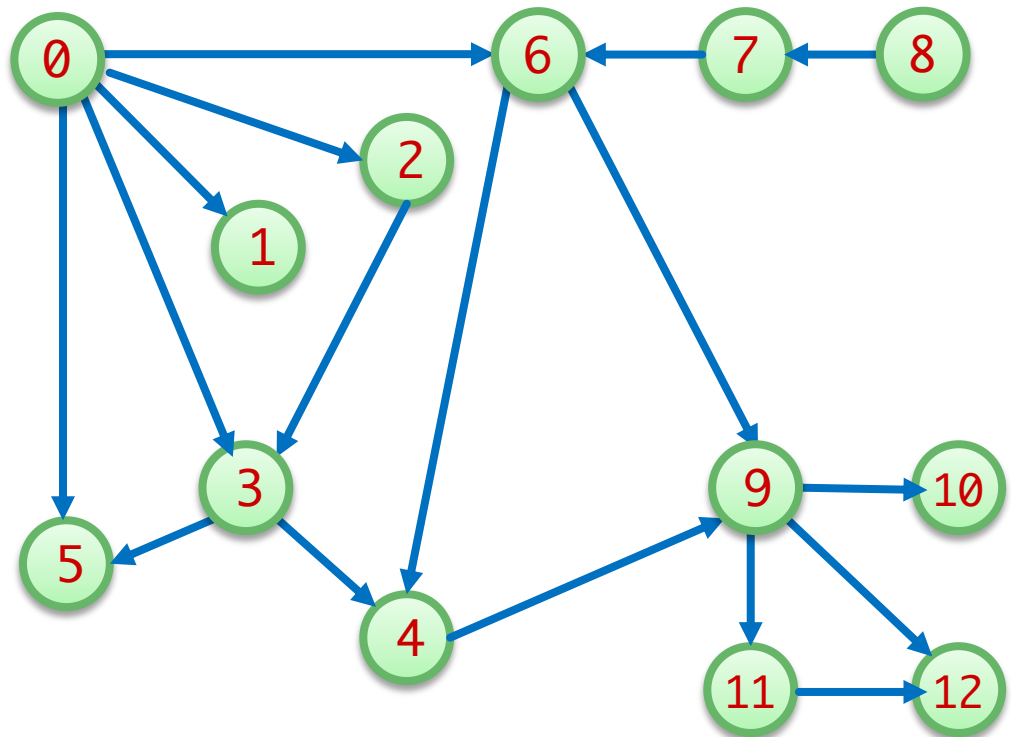
8 -> 0

9 -> 2

10 -> 1

11 -> 1

12 -> 2



# Orden Topológico en DiGrafos

Seleccionamos fuentes  $F = \{0, 8\}$

vértice  $\rightarrow$  # predecesores pendientes

**0  $\rightarrow$  0**

1  $\rightarrow$  1

2  $\rightarrow$  1

3  $\rightarrow$  2

4  $\rightarrow$  2

5  $\rightarrow$  2

6  $\rightarrow$  2

7  $\rightarrow$  1

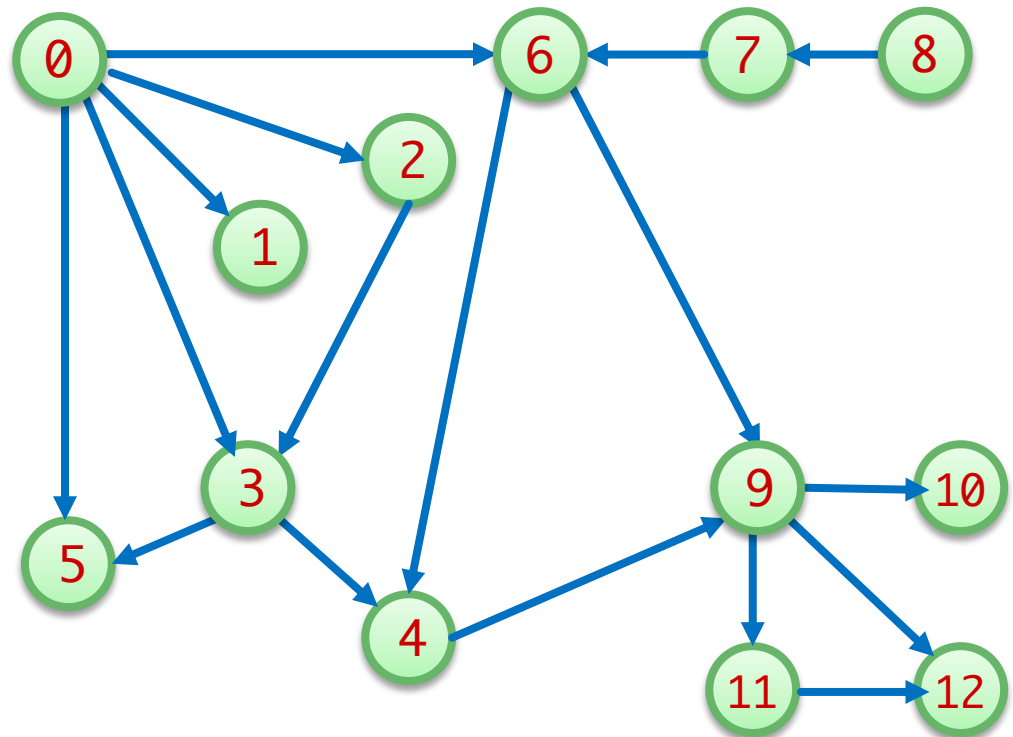
**8  $\rightarrow$  0**

9  $\rightarrow$  2

10  $\rightarrow$  1

11  $\rightarrow$  1

12  $\rightarrow$  2

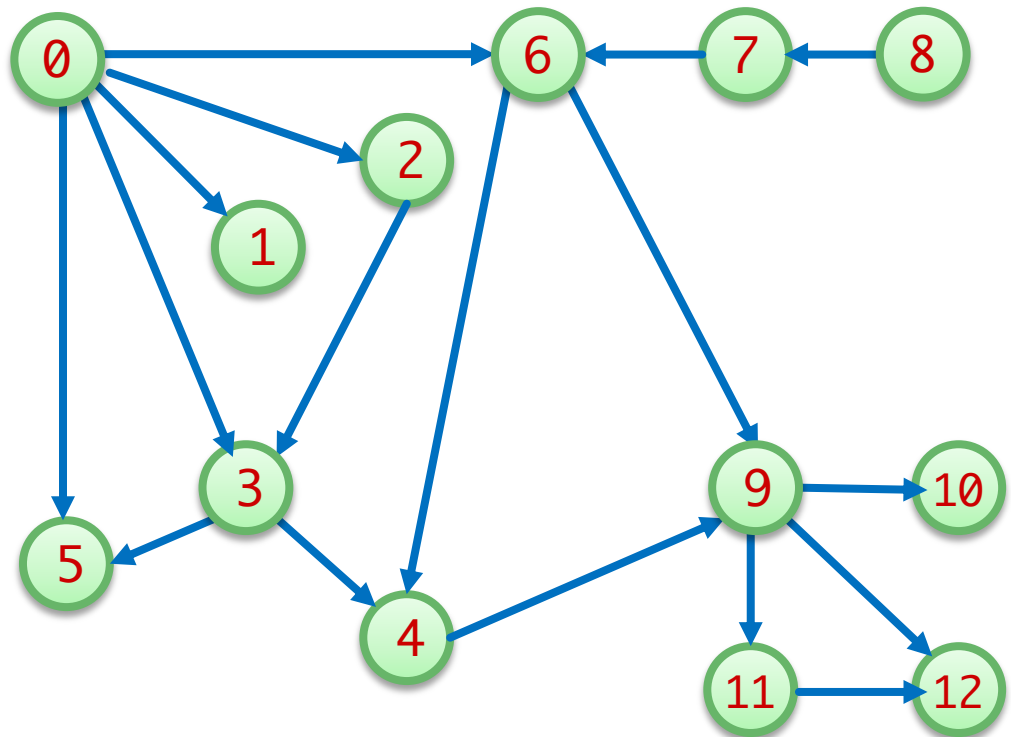


# Orden Topológico en DiGrafos

Eliminamos fuentes  $F = \{0, 8\}$  del diccionario

**vértice -> # predecesores pendientes**

1 -> 1  
2 -> 1  
3 -> 2  
4 -> 2  
5 -> 2  
6 -> 2  
7 -> 1  
9 -> 2  
10 -> 1  
11 -> 1  
12 -> 2



# Orden Topológico en DiGrafos

Añadimos fuentes  $F = \{0, 8\}$  al orden topológico

vértice -> # predecesores pendientes

1 -> 1

2 -> 1

3 -> 2

4 -> 2

5 -> 2

6 -> 2

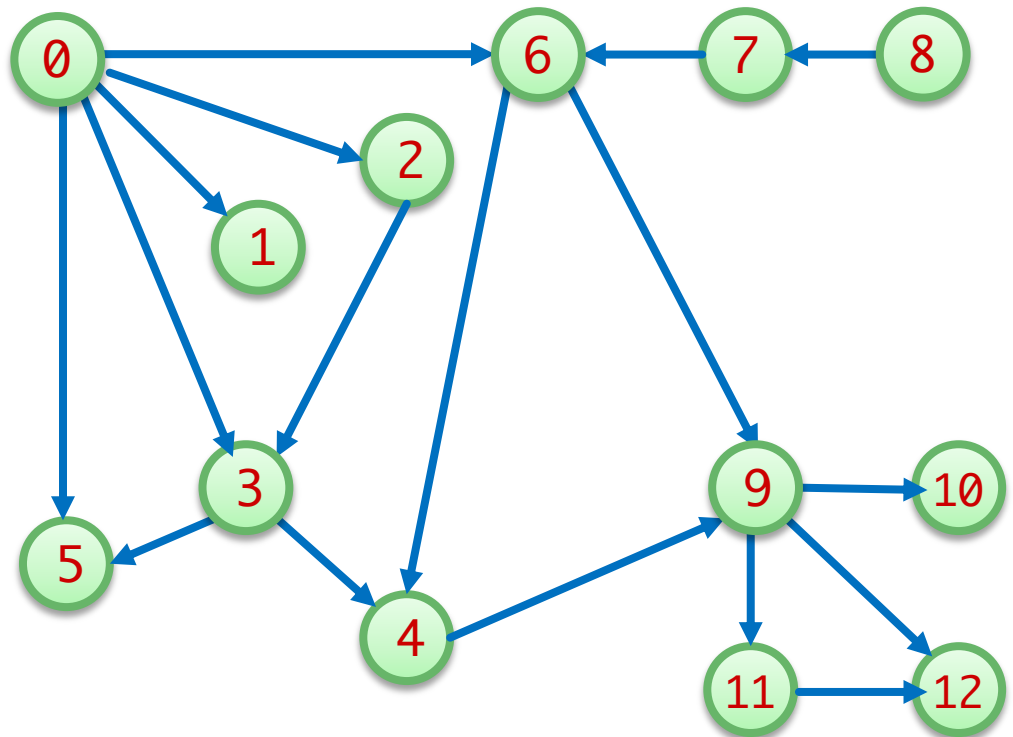
7 -> 1

9 -> 2

10 -> 1

11 -> 1

12 -> 2



Orden Topológico:  $8 < 0$

# Orden Topológico en DiGrafos

Restamos fuentes  $F = \{0, 8\}$  del diccionario

vértice -> # predecesores pendientes

1 -> 0

2 -> 0

3 -> 1

4 -> 2

5 -> 1

6 -> 1

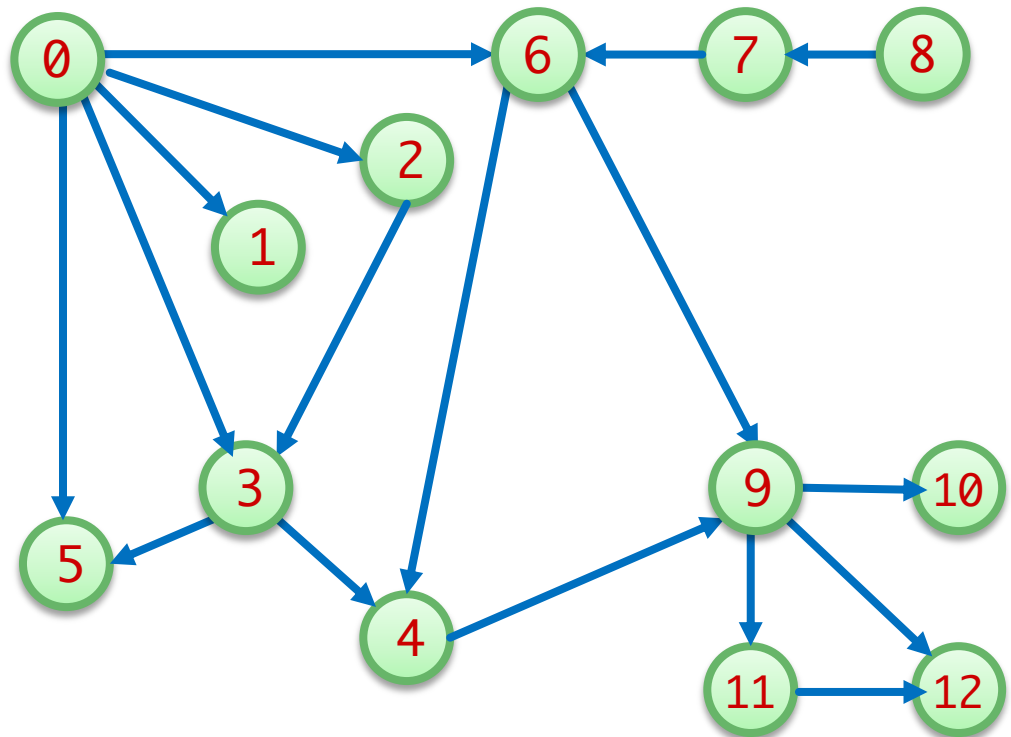
7 -> 0

9 -> 2

10 -> 1

11 -> 1

12 -> 2



Orden Topológico:  $8 < 0$

# Orden Topológico en DiGrafos

Seleccionamos fuentes  $F = \{1, 2, 7\}$

vértice  $\rightarrow$  # predecesores pendientes

1  $\rightarrow$  0

2  $\rightarrow$  0

3  $\rightarrow$  1

4  $\rightarrow$  2

5  $\rightarrow$  1

6  $\rightarrow$  1

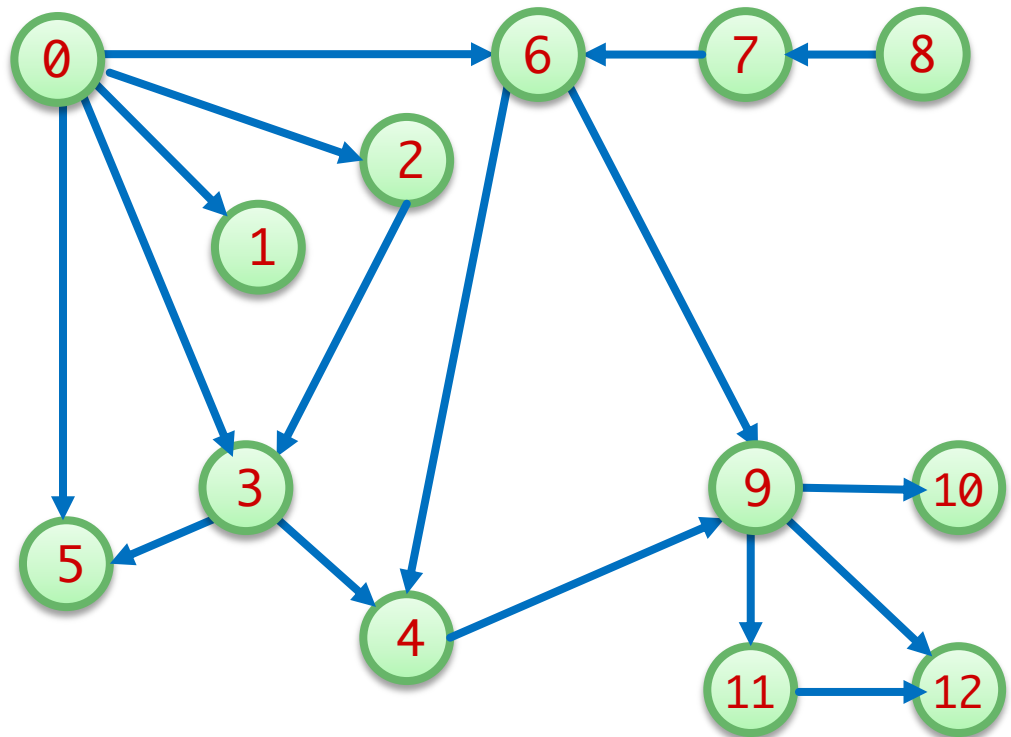
7  $\rightarrow$  0

9  $\rightarrow$  2

10  $\rightarrow$  1

11  $\rightarrow$  1

12  $\rightarrow$  2



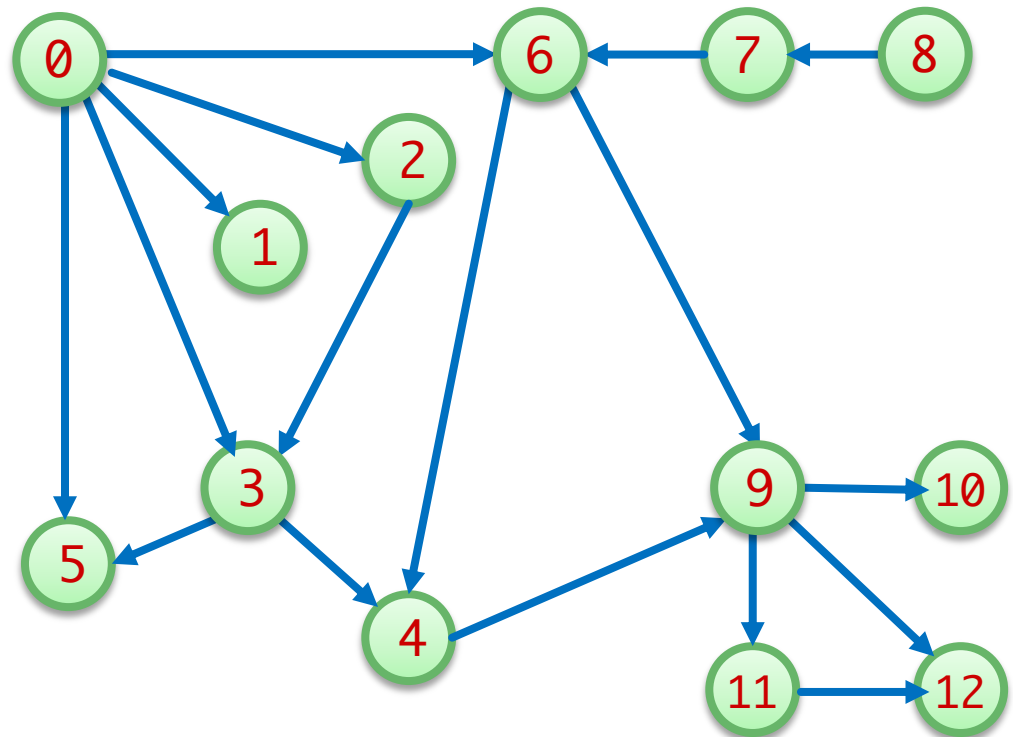
Orden Topológico: 8  $\prec$  0

# Orden Topológico en DiGrafos

Eliminamos fuentes  $F = \{1, 2, 7\}$  del diccionario

**vértice -> # predecesores pendientes**

3 -> 1  
4 -> 2  
5 -> 1  
6 -> 1  
9 -> 2  
10 -> 1  
11 -> 1  
12 -> 2



Orden Topológico:  $8 < 0$

# Orden Topológico en DiGrafos

Añadimos fuentes  $F = \{1, 2, 7\}$  al orden topológico

vértice  $\rightarrow$  # predecesores pendientes

3  $\rightarrow$  1

4  $\rightarrow$  2

5  $\rightarrow$  1

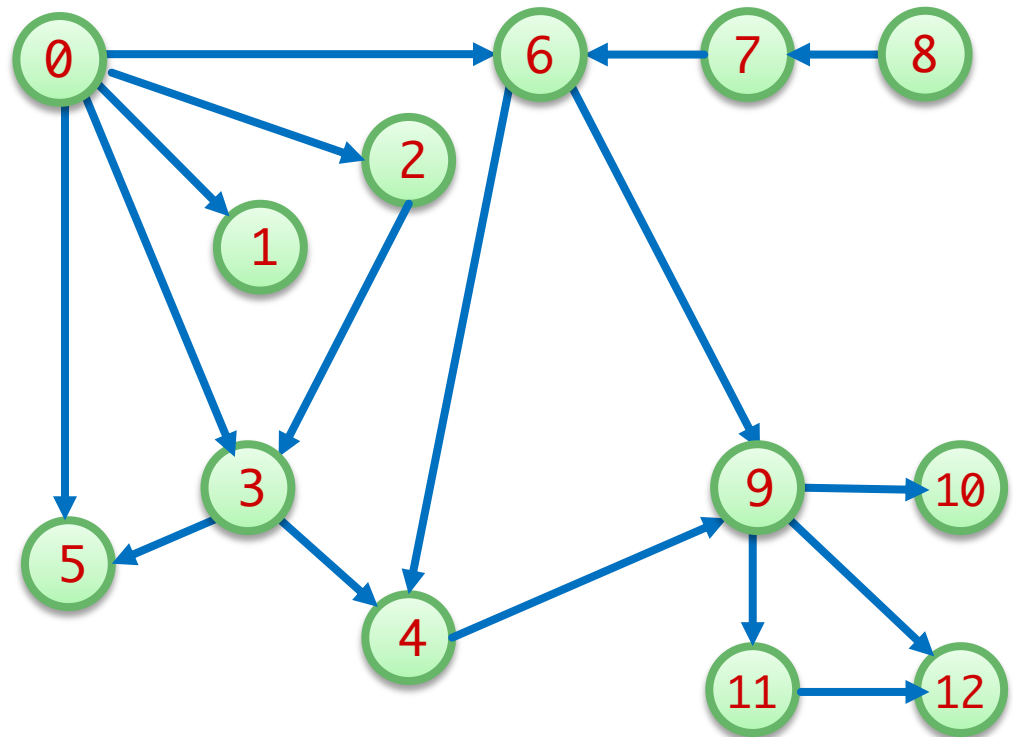
6  $\rightarrow$  1

9  $\rightarrow$  2

10  $\rightarrow$  1

11  $\rightarrow$  1

12  $\rightarrow$  2



Orden Topológico: 8  $\prec$  0  $\prec$  2  $\prec$  7  $\prec$  1



# Orden Topológico en DiGrafos

Restamos fuentes  $F = \{1, 2, 7\}$  del diccionario

**vértice -> # predecesores pendientes**

3 -> 0

4 -> 2

5 -> 1

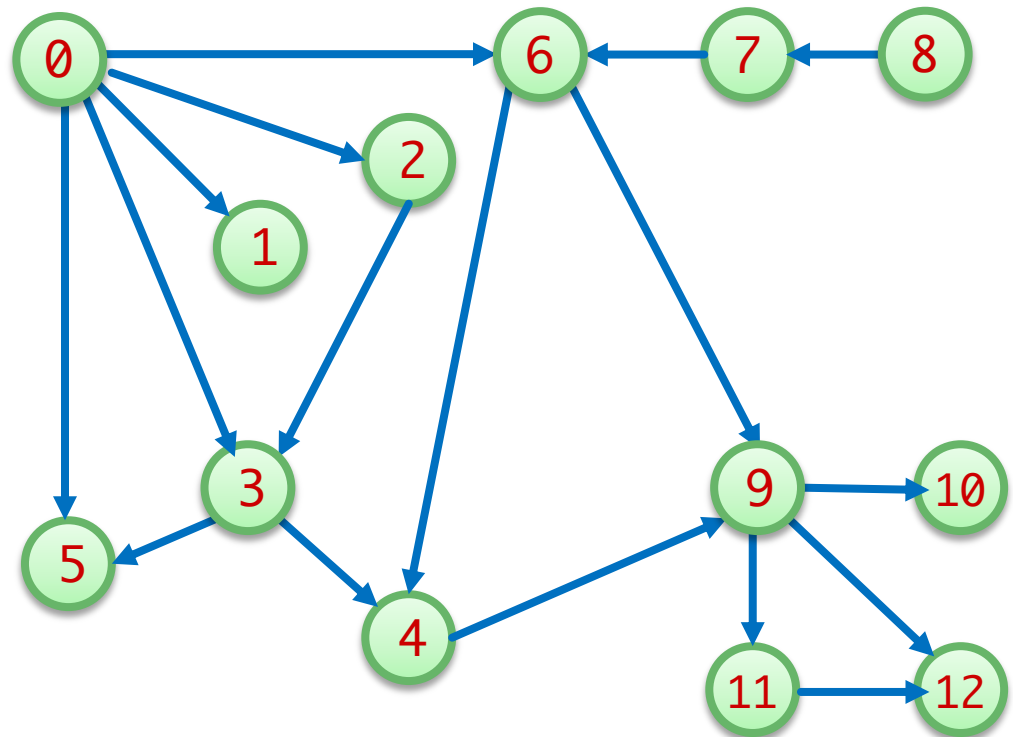
6 -> 0

9 -> 2

10 -> 1

11 -> 1

12 -> 2



Orden Topológico: 8 < 0 < 2 < 7 < 1

# Orden Topológico en DiGrafos

Seleccionamos fuentes  $F = \{3, 6\}$  del diccionario

vértice  $\rightarrow$  # predecesores pendientes

3  $\rightarrow$  0

4  $\rightarrow$  2

5  $\rightarrow$  1

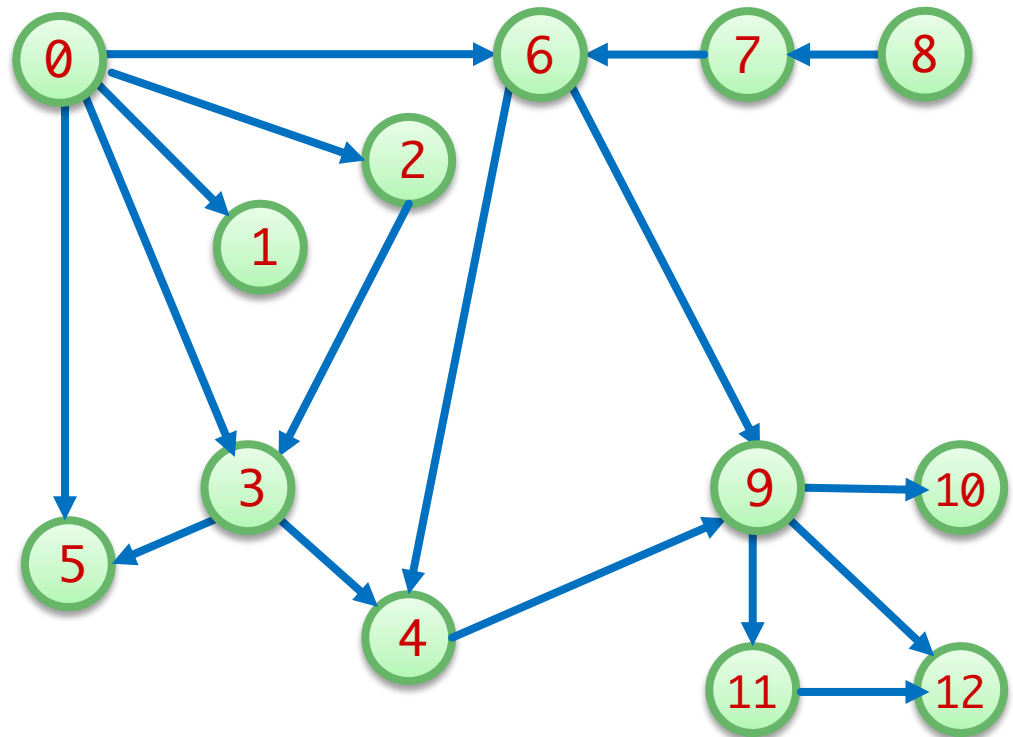
6  $\rightarrow$  0

9  $\rightarrow$  2

10  $\rightarrow$  1

11  $\rightarrow$  1

12  $\rightarrow$  2



Orden Topológico: 8  $\prec$  0  $\prec$  2  $\prec$  7  $\prec$  1

# Orden Topológico en DiGrafos

Eliminamos fuentes  $F = \{3, 6\}$  del diccionario

**vértice -> # predecesores pendientes**

4 -> 2

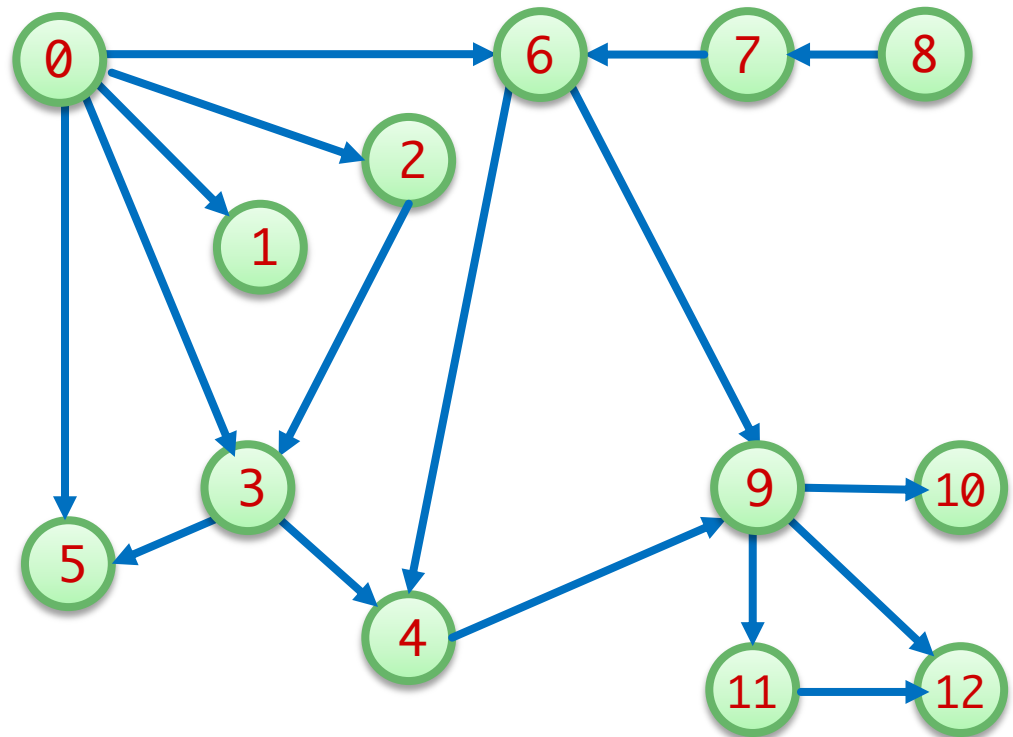
5 -> 1

9 -> 2

10 -> 1

11 -> 1

12 -> 2



Orden Topológico:  $8 \prec 0 \prec 2 \prec 7 \prec 1$

# Orden Topológico en DiGrafos

Añadimos fuentes  $F = \{3, 6\}$  al orden topológico

vértice  $\rightarrow$  # predecesores pendientes

4  $\rightarrow$  2

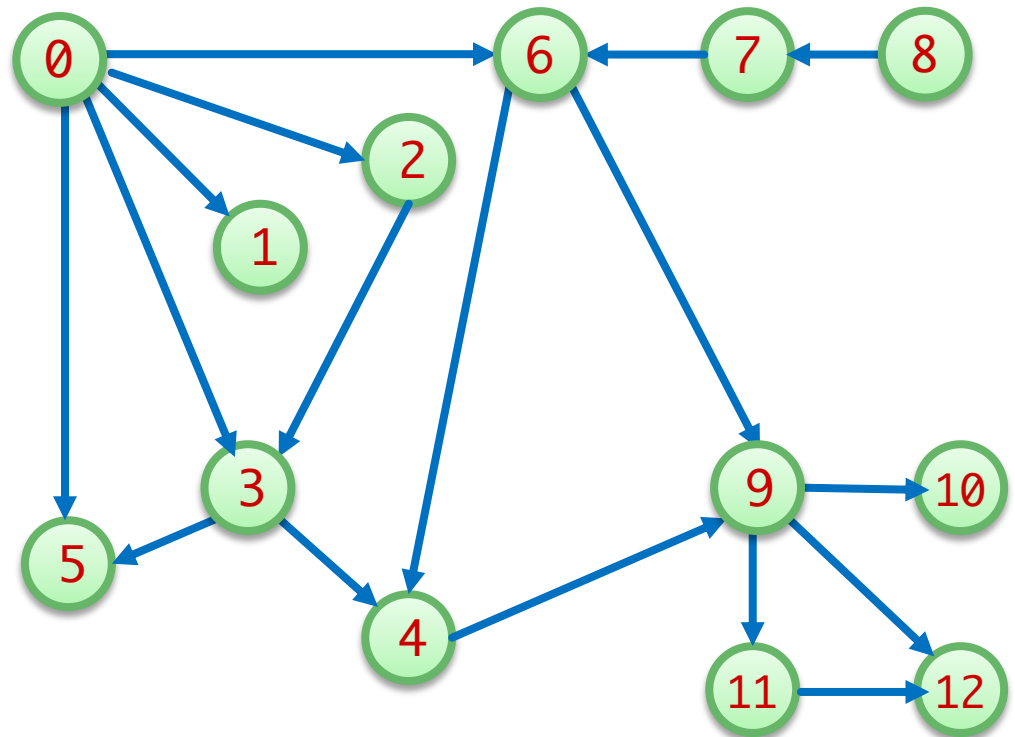
5  $\rightarrow$  1

9  $\rightarrow$  2

10  $\rightarrow$  1

11  $\rightarrow$  1

12  $\rightarrow$  2



Orden Topológico: 8 < 0 < 2 < 7 < 1 < 3 < 6

# Orden Topológico en DiGrafos

Restamos fuentes  $F = \{3, 6\}$  del diccionario

vértice -> # predecesores pendientes

4 -> 0

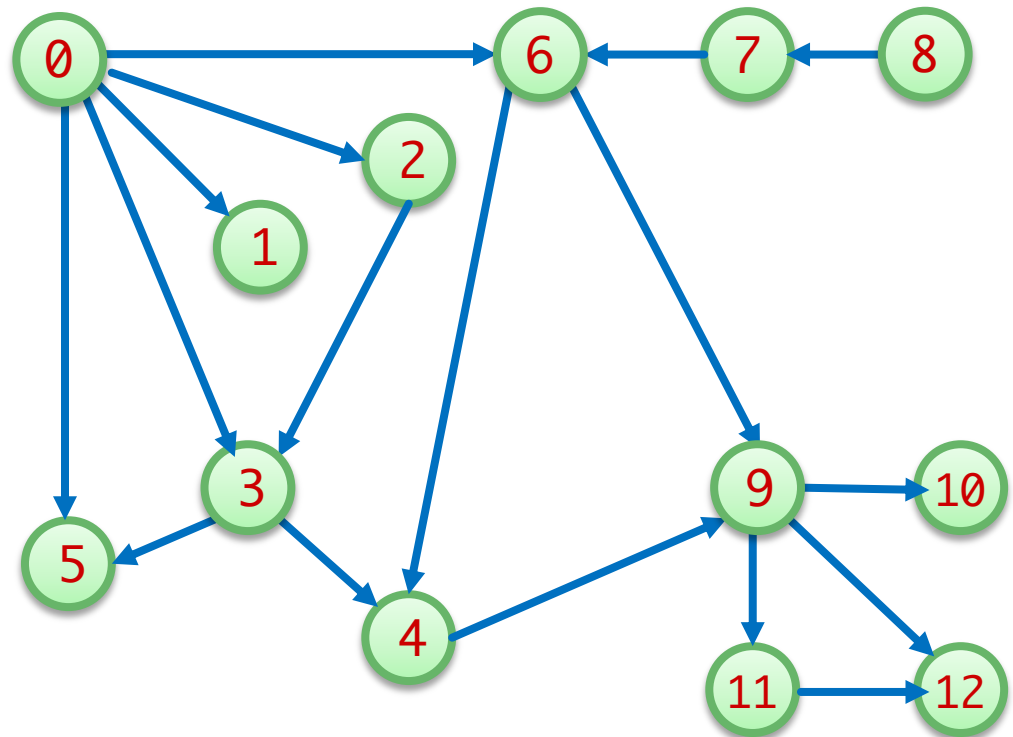
5 -> 0

9 -> 1

10 -> 1

11 -> 1

12 -> 2



Orden Topológico: 8 < 0 < 2 < 7 < 1 < 3 < 6

# Orden Topológico en DiGrafos

Seleccionamos fuentes  $F = \{4, 5\}$  del diccionario

vértice  $\rightarrow$  # predecesores pendientes

4  $\rightarrow$  0

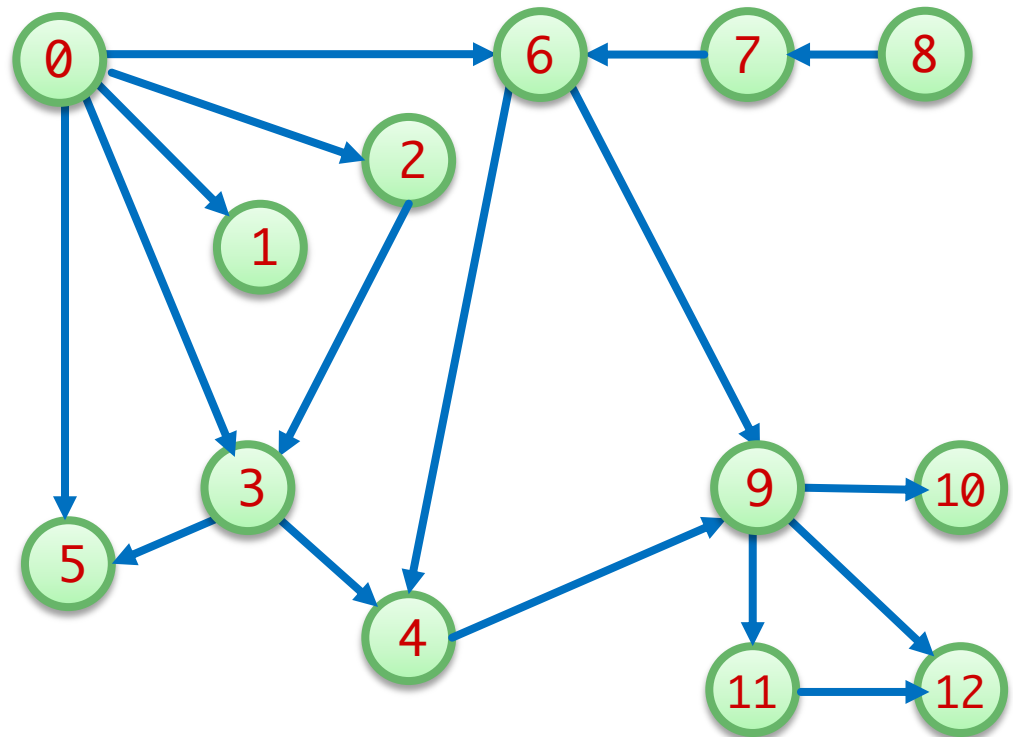
5  $\rightarrow$  0

9  $\rightarrow$  1

10  $\rightarrow$  1

11  $\rightarrow$  1

12  $\rightarrow$  2



Orden Topológico: 8  $\prec$  0  $\prec$  2  $\prec$  7  $\prec$  1  $\prec$  3  $\prec$  6

# Orden Topológico en DiGrafos

Eliminamos fuentes  $F = \{4, 5\}$  del diccionario

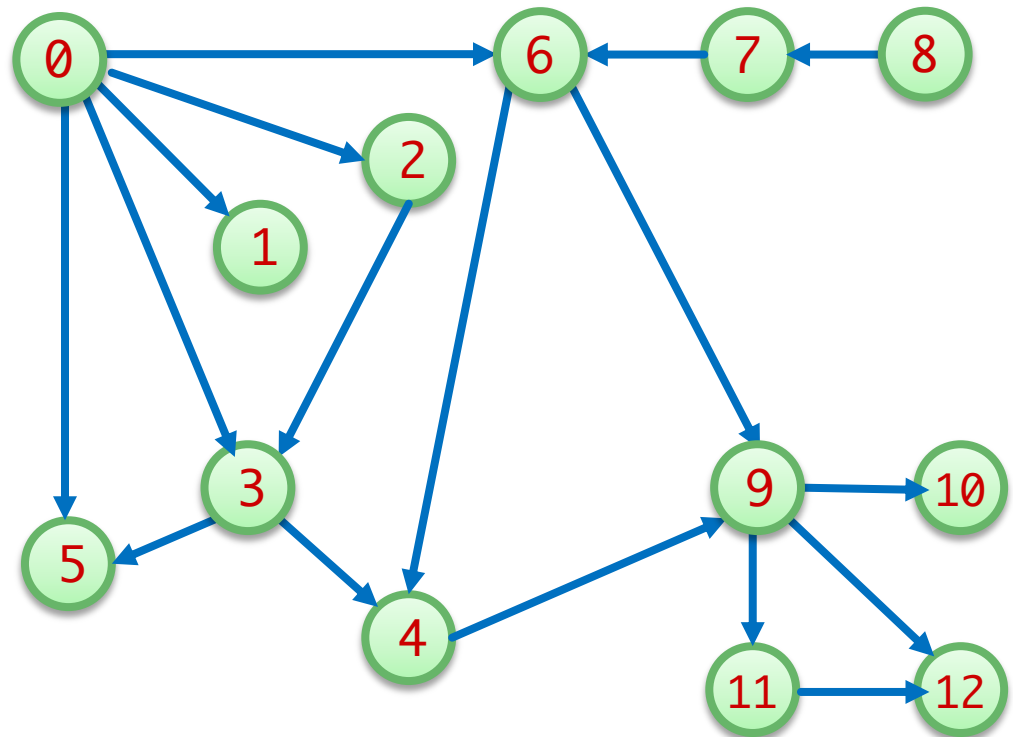
**vértice -> # predecesores pendientes**

9 -> 1

10 -> 1

11 -> 1

12 -> 2



Orden Topológico:  $8 \prec 0 \prec 2 \prec 7 \prec 1 \prec 3 \prec 6$

# Orden Topológico en DiGrafos

Añadimos fuentes  $F = \{4, 5\}$  al orden topológico

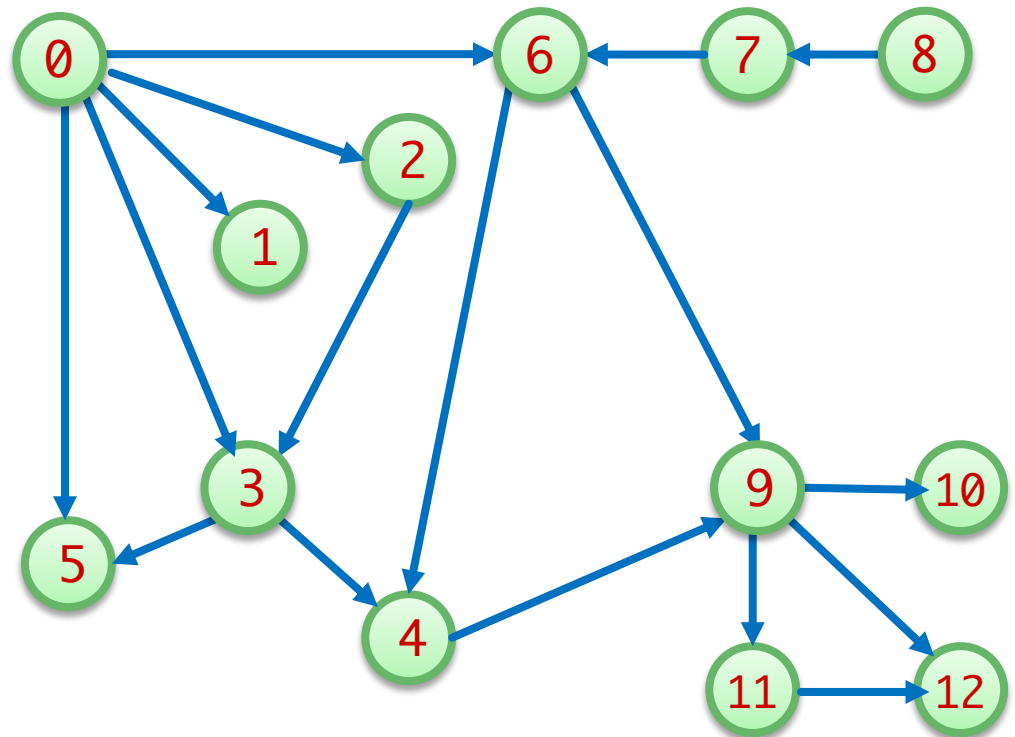
vértice -> # predecesores pendientes

9 -> 1

10 -> 1

11 -> 1

12 -> 2



Orden Topológico: 8 < 0 < 2 < 7 < 1 < 3 < 6 < 4 < 5



# Orden Topológico en DiGrafos

Restamos fuentes  $F = \{4, 5\}$  del diccionario

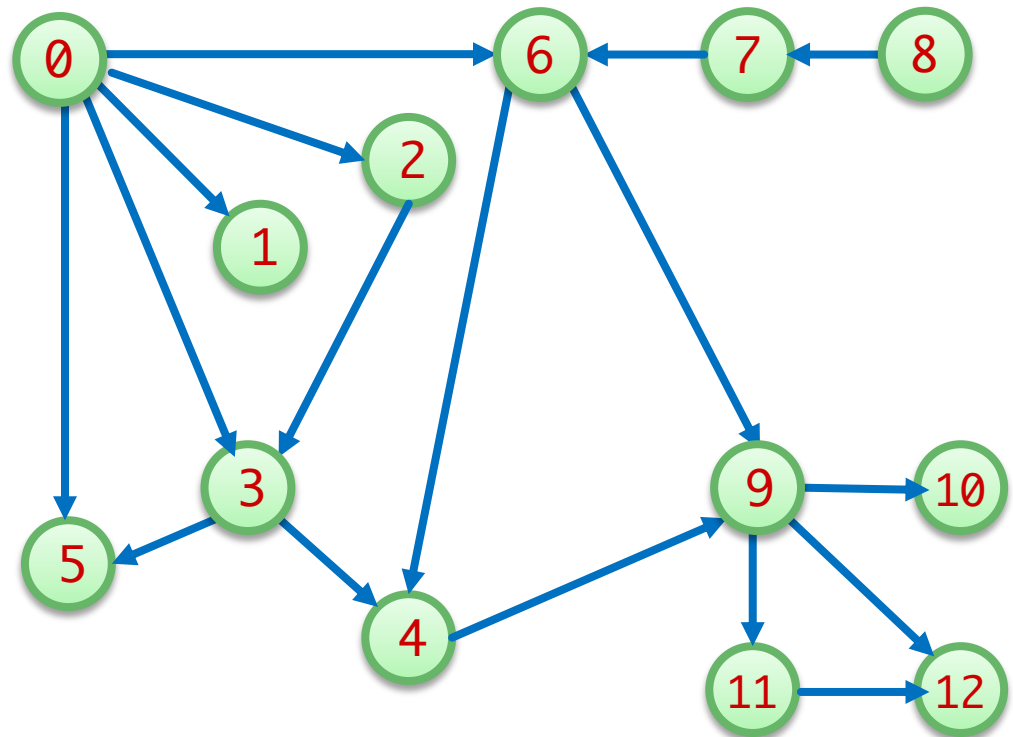
vértice -> # predecesores pendientes

9 -> 0

10 -> 1

11 -> 1

12 -> 2



Orden Topológico: 8 < 0 < 2 < 7 < 1 < 3 < 6 < 4 < 5

# Orden Topológico en DiGrafos

Seleccionamos fuentes  $F = \{9\}$  del diccionario

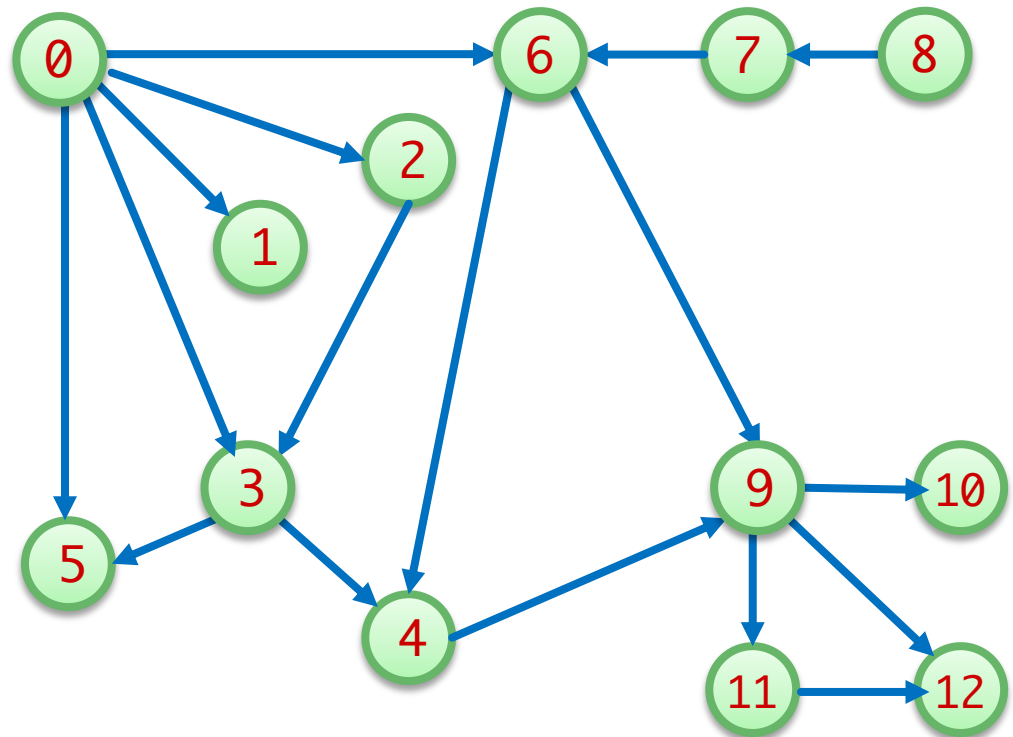
vértice  $\rightarrow$  # predecesores pendientes

9  $\rightarrow$  0

10  $\rightarrow$  1

11  $\rightarrow$  1

12  $\rightarrow$  2



Orden Topológico: 8  $\prec$  0  $\prec$  2  $\prec$  7  $\prec$  1  $\prec$  3  $\prec$  6  $\prec$  4  $\prec$  5

# Orden Topológico en DiGrafos

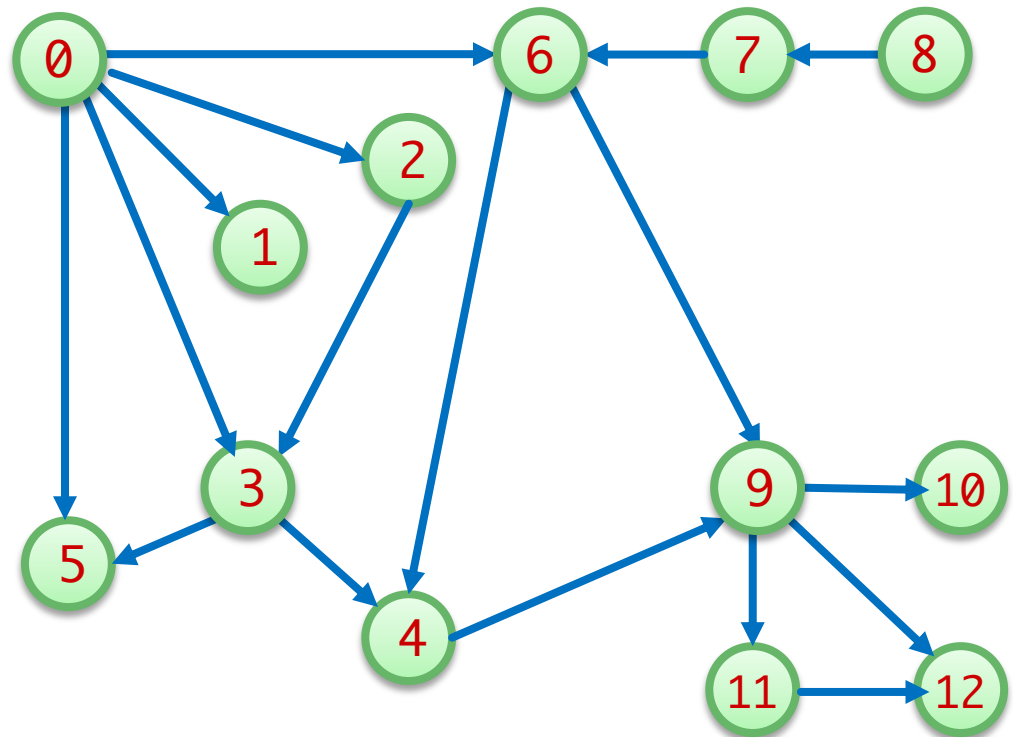
Eliminamos fuentes  $F = \{9\}$  del diccionario

**vértice -> # predecesores pendientes**

10 -> 1

11 -> 1

12 -> 2



Orden Topológico: 8 < 0 < 2 < 7 < 1 < 3 < 6 < 4 < 5

# Orden Topológico en DiGrafos

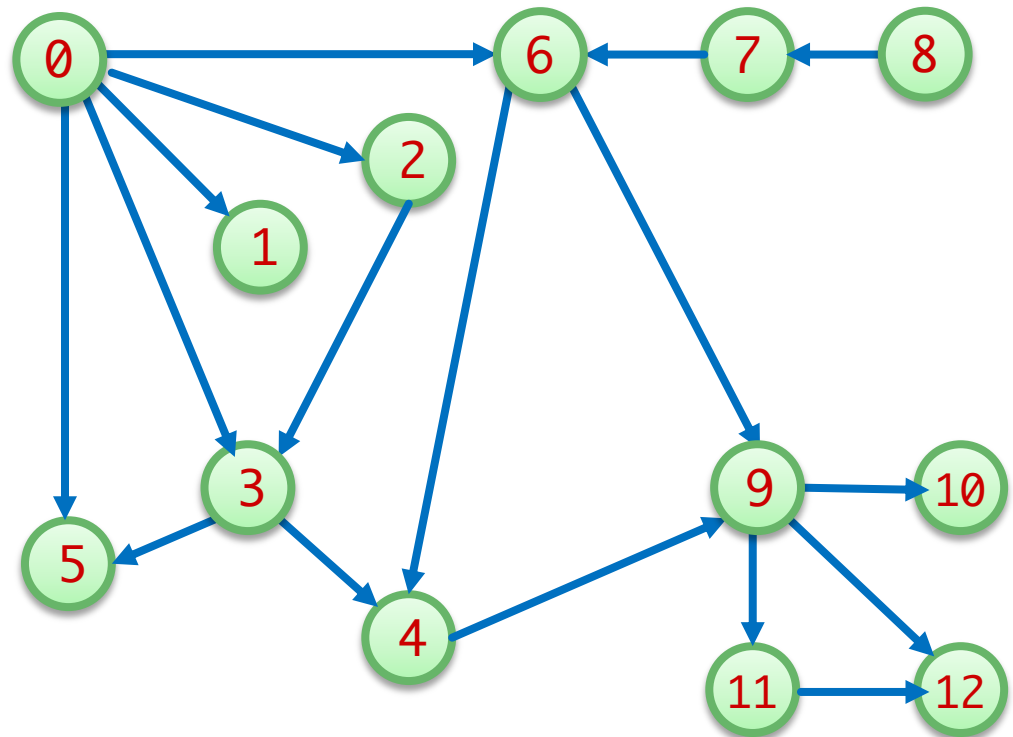
Añadimos fuentes  $F = \{9\}$  al orden topológico

**vértice -> # predecesores pendientes**

10 -> 1

11 -> 1

12 -> 2



Orden Topológico: 8 < 0 < 2 < 7 < 1 < 3 < 6 < 4 < 5 < 9

# Orden Topológico en DiGrafos

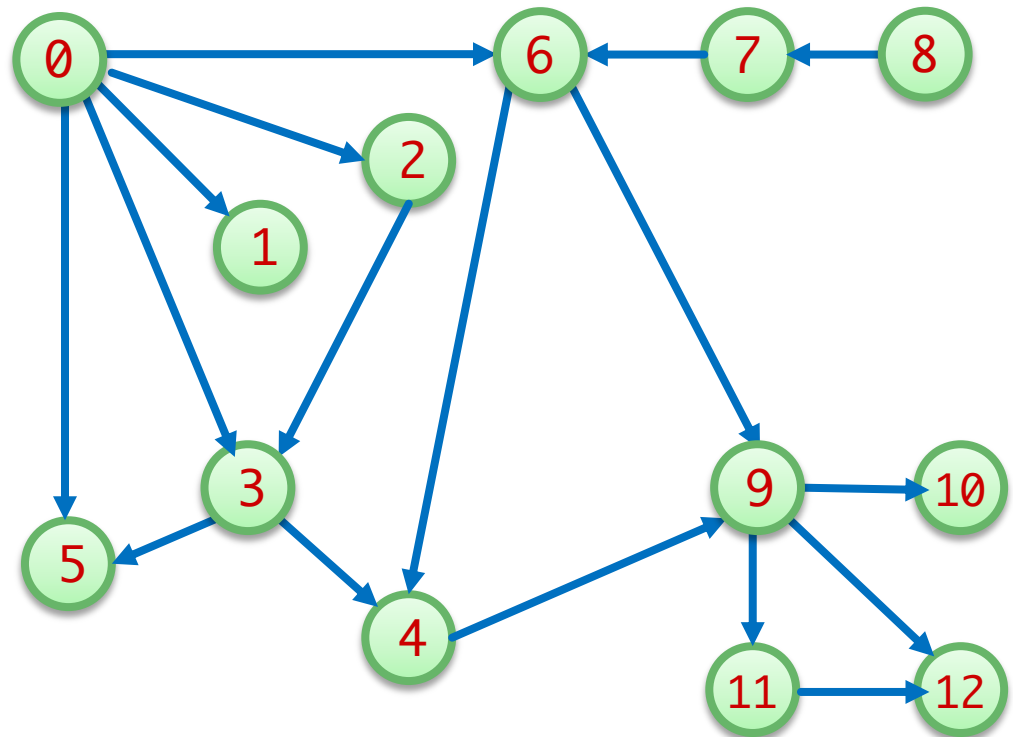
Restamos fuentes  $F = \{9\}$  del diccionario

vértice  $\rightarrow$  # predecesores pendientes

10  $\rightarrow$  0

11  $\rightarrow$  0

12  $\rightarrow$  1



Orden Topológico: 8  $\prec$  0  $\prec$  2  $\prec$  7  $\prec$  1  $\prec$  3  $\prec$  6  $\prec$  4  $\prec$  5  $\prec$  9

# Orden Topológico en DiGrafos

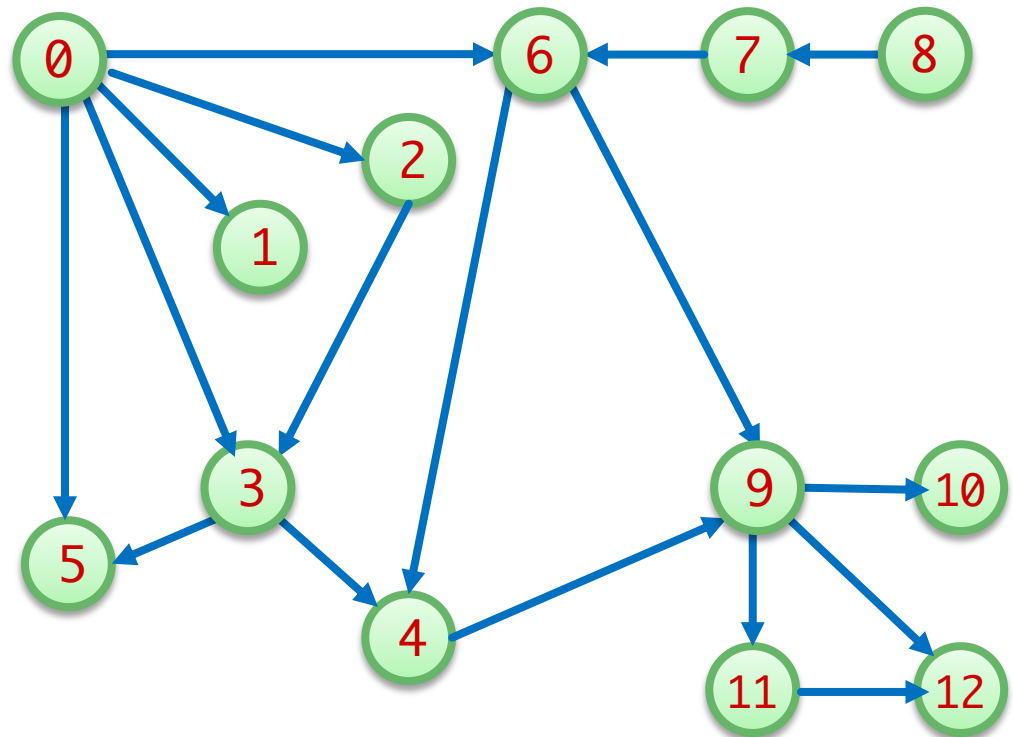
Seleccionamos fuentes  $F = \{10, 11\}$  del diccionario

vértice  $\rightarrow$  # predecesores pendientes

10  $\rightarrow$  0

11  $\rightarrow$  0

12  $\rightarrow$  1



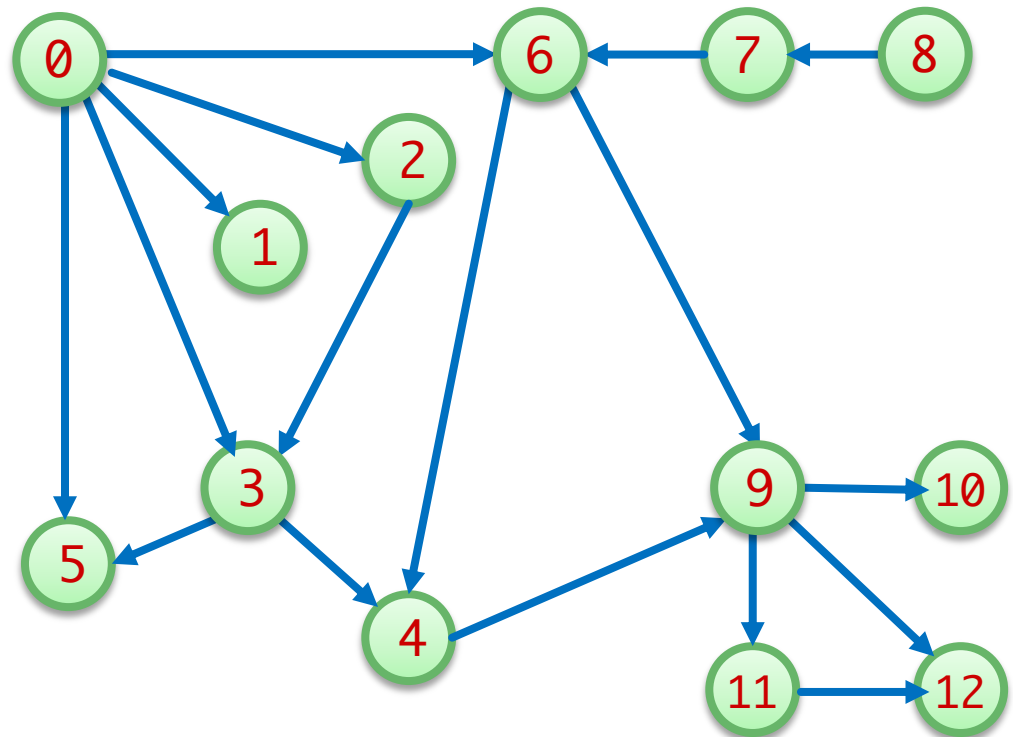
Orden Topológico: 8  $\prec$  0  $\prec$  2  $\prec$  7  $\prec$  1  $\prec$  3  $\prec$  6  $\prec$  4  $\prec$  5  $\prec$  9

# Orden Topológico en DiGrafos

Eliminamos fuentes  $F = \{10, 11\}$  del diccionario

vértice  $\rightarrow$  # predecesores pendientes

12  $\rightarrow$  1



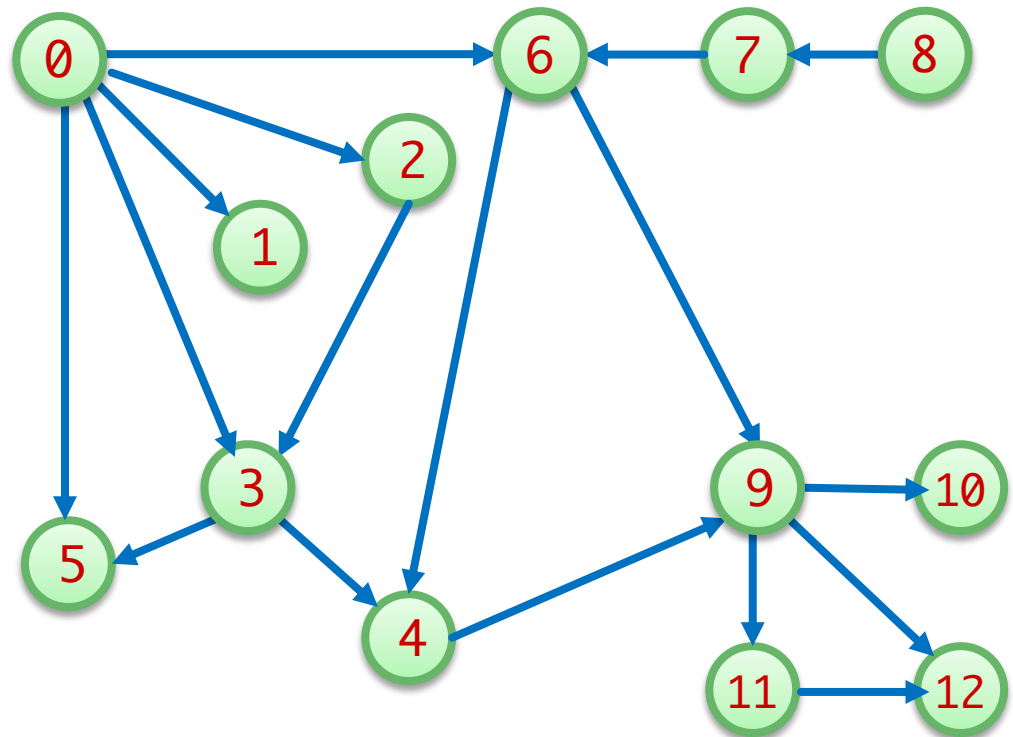
Orden Topológico: 8  $\prec$  0  $\prec$  2  $\prec$  7  $\prec$  1  $\prec$  3  $\prec$  6  $\prec$  4  $\prec$  5  $\prec$  9

# Orden Topológico en DiGrafos

Añadimos fuentes  $F = \{10, 11\}$  al orden topológico

vértice  $\rightarrow$  # predecesores pendientes

12  $\rightarrow$  1



Orden Topológico: 8 < 0 < 2 < 7 < 1 < 3 < 6 < 4 < 5 < 9 < 10 < 11

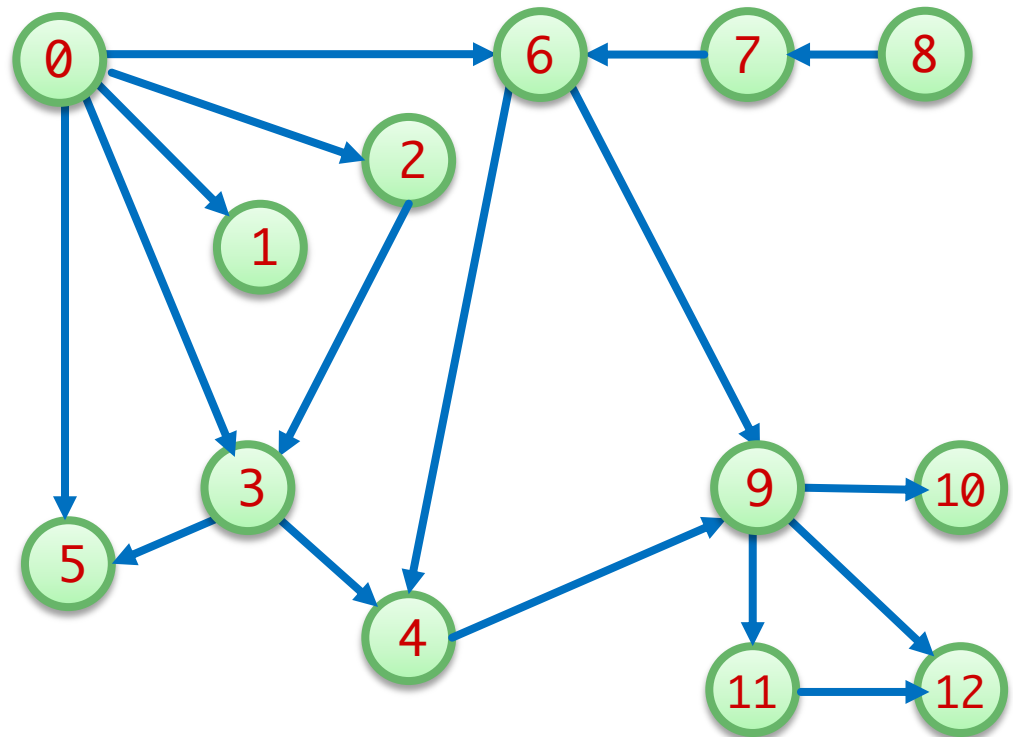


# Orden Topológico en DiGrafos

Restamos fuentes  $F = \{10, 11\}$  del diccionario

vértice  $\rightarrow$  # predecesores pendientes

12  $\rightarrow$  0



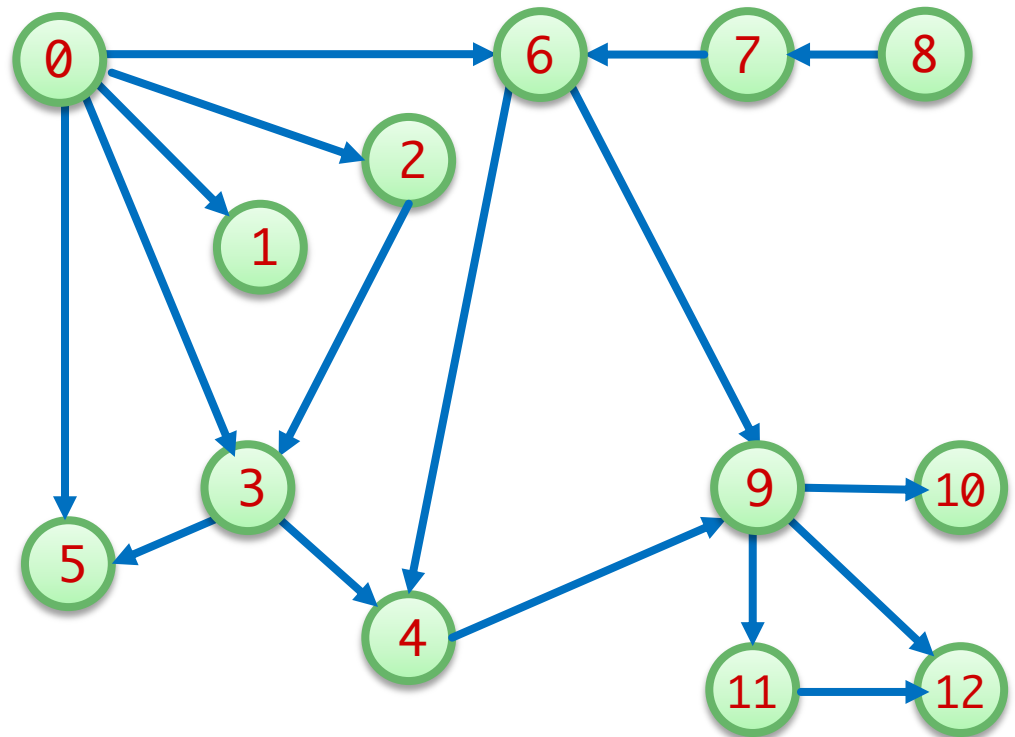
Orden Topológico: 8 < 0 < 2 < 7 < 1 < 3 < 6 < 4 < 5 < 9 < 10 < 11

# Orden Topológico en DiGrafos

Seleccionamos fuentes  $F = \{12\}$  del diccionario

vértice  $\rightarrow$  # predecesores pendientes

12  $\rightarrow$  0

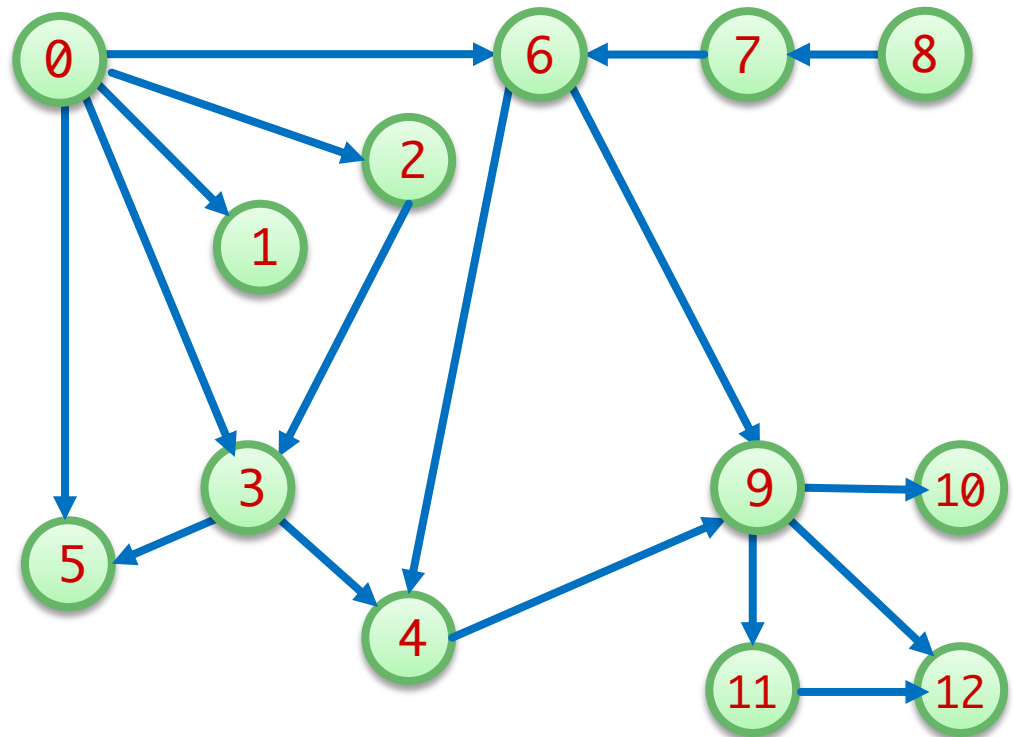


Orden Topológico: 8  $\prec$  0  $\prec$  2  $\prec$  7  $\prec$  1  $\prec$  3  $\prec$  6  $\prec$  4  $\prec$  5  $\prec$  9  $\prec$  10  $\prec$  11

# Orden Topológico en DiGrafos

Eliminamos fuentes  $F = \{12\}$  del diccionario

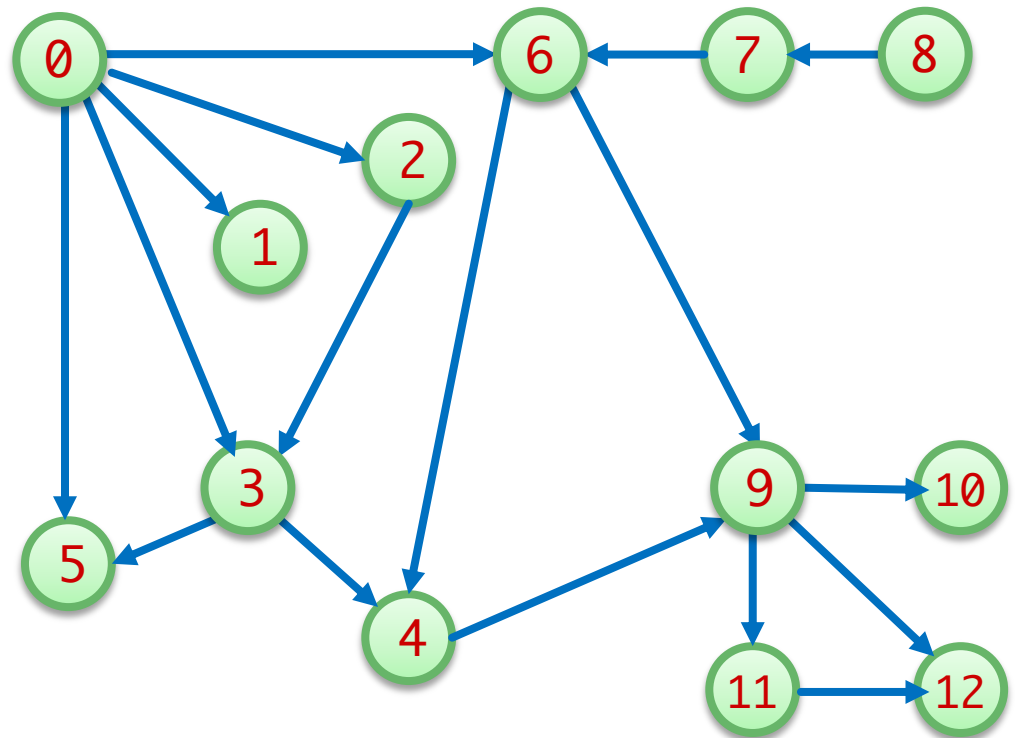
vértice  $\rightarrow$  # predecesores pendientes



Orden Topológico:  $8 \prec 0 \prec 2 \prec 7 \prec 1 \prec 3 \prec 6 \prec 4 \prec 5 \prec 9 \prec 10 \prec 11$

# Orden Topológico en DiGrafos

Añadimos fuentes  $F = \{12\}$  al orden topológico  
vértice  $\rightarrow$  # predecesores pendientes

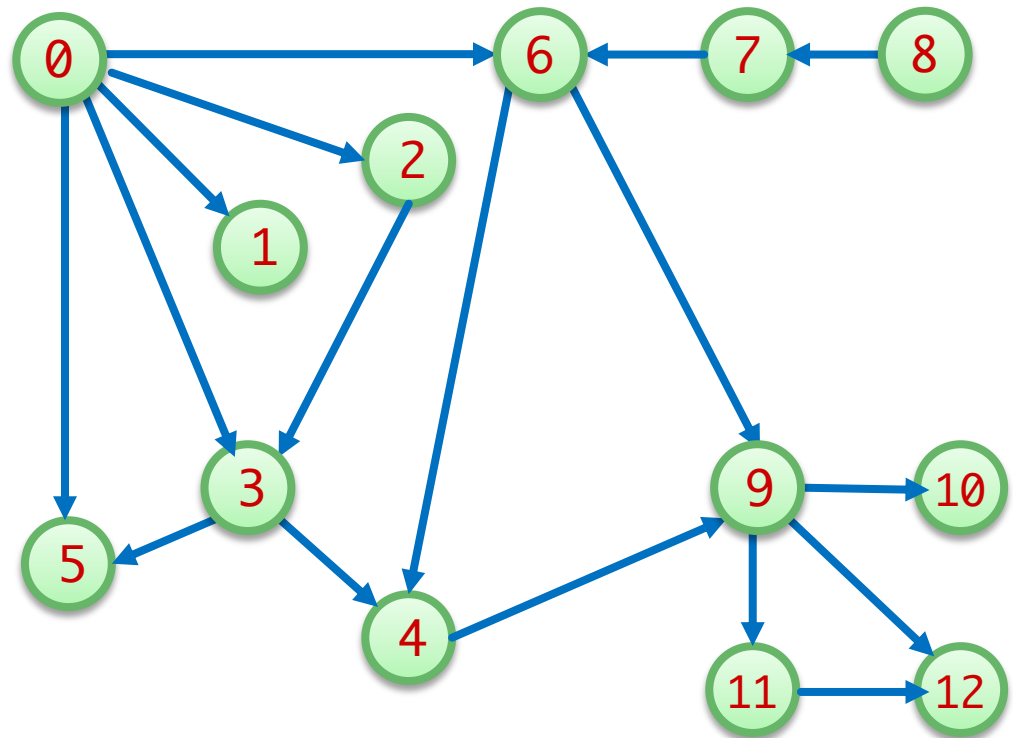


Orden Topológico: 8  $\prec$  0  $\prec$  2  $\prec$  7  $\prec$  1  $\prec$  3  $\prec$  6  $\prec$  4  $\prec$  5  $\prec$  9  $\prec$  10  $\prec$  11  $\prec$  12

# Orden Topológico en DiGrafos

Restamos fuentes  $F = \{12\}$  del diccionario

vértice  $\rightarrow$  # predecesores pendientes

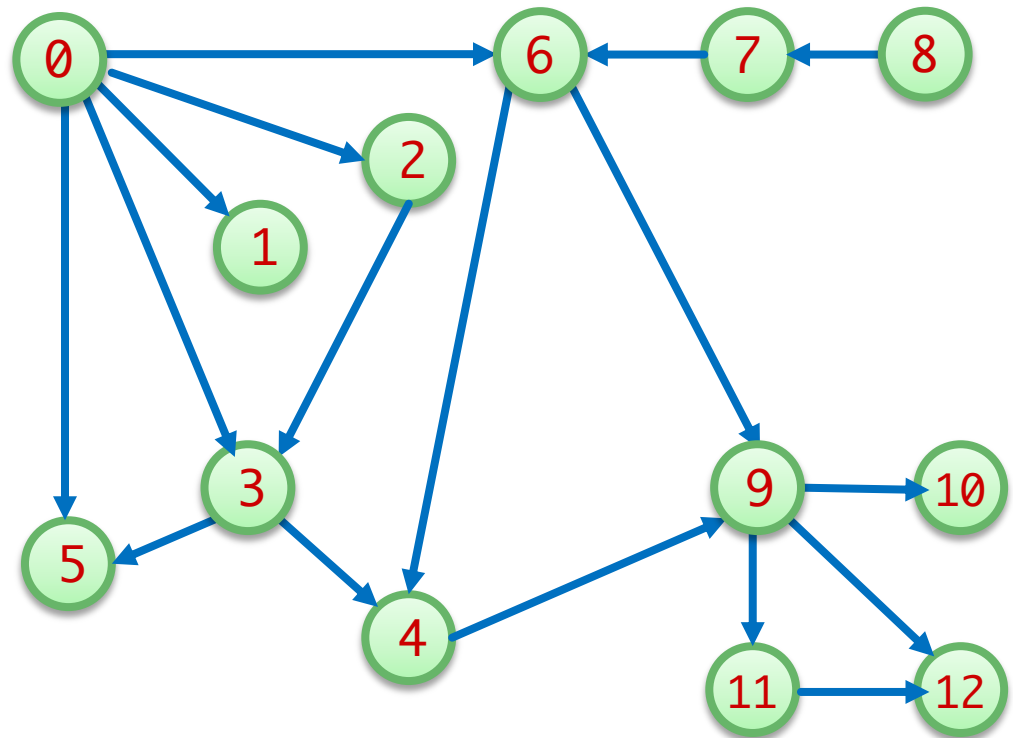


Orden Topológico:  $8 \prec 0 \prec 2 \prec 7 \prec 1 \prec 3 \prec 6 \prec 4 \prec 5 \prec 9 \prec 10 \prec 11 \prec 12$

# Orden Topológico en DiGrafos

El diccionario está vacío: hemos terminado

vértice -> # predecesores pendientes



Orden Topológico: 8 < 0 < 2 < 7 < 1 < 3 < 6 < 4 < 5 < 9 < 10 < 11 < 12

# Orden Topológico. DiGrafo cíclico

Inicializamos diccionario con grados de entrada

**vértice -> # predecesores pendientes**

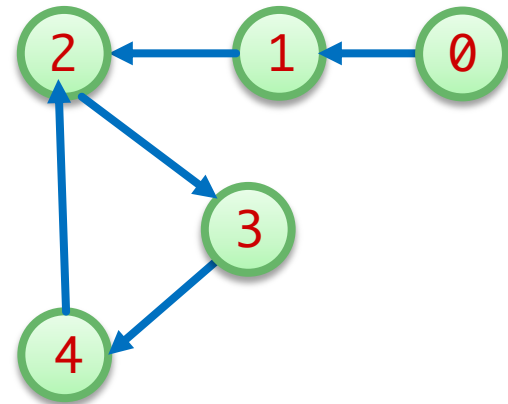
0 -> 0

1 -> 1

2 -> 2

3 -> 1

4 -> 1



# Orden Topológico. DiGrafo cíclico

Seleccionamos fuentes  $F = \{0\}$  del diccionario

vértice -> # predecesores pendientes

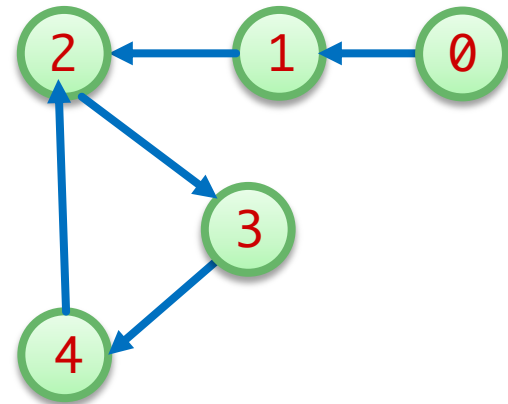
0 -> 0

1 -> 1

2 -> 2

3 -> 1

4 -> 1





# Orden Topológico. DiGrafo cíclico

Eliminamos fuentes  $F = \{0\}$  del diccionario

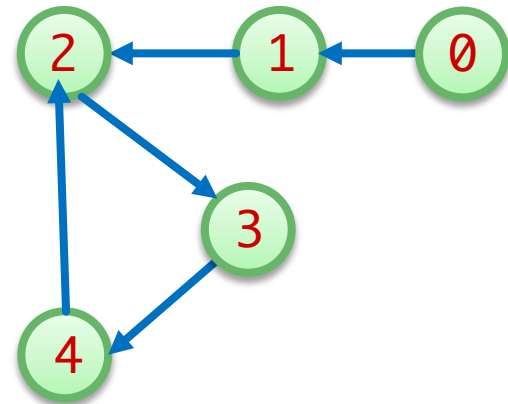
**vértice -> # predecesores pendientes**

1 -> 1

2 -> 2

3 -> 1

4 -> 1



# Orden Topológico. DiGrafo cíclico

Añadimos fuentes  $F = \{0\}$  al orden topológico

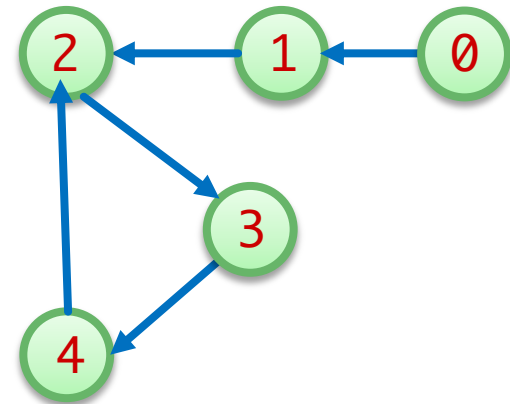
**vértice -> # predecesores pendientes**

1 -> 1

2 -> 2

3 -> 1

4 -> 1



Orden Topológico: 0

# Orden Topológico. DiGrafo cíclico

Restamos fuentes  $F = \{0\}$  del diccionario

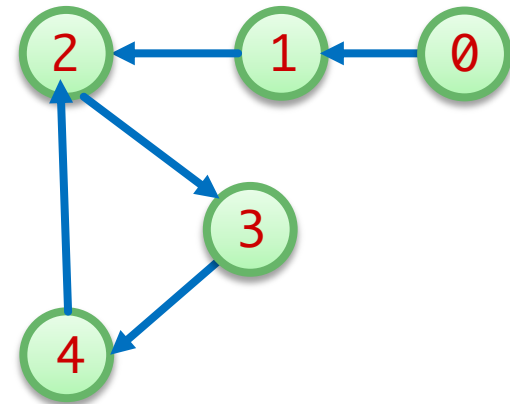
**vértice -> # predecesores pendientes**

1 -> 0

2 -> 2

3 -> 1

4 -> 1



Orden Topológico: 0

# Orden Topológico. DiGrafo cíclico

Seleccionamos fuentes  $F = \{1\}$  del diccionario

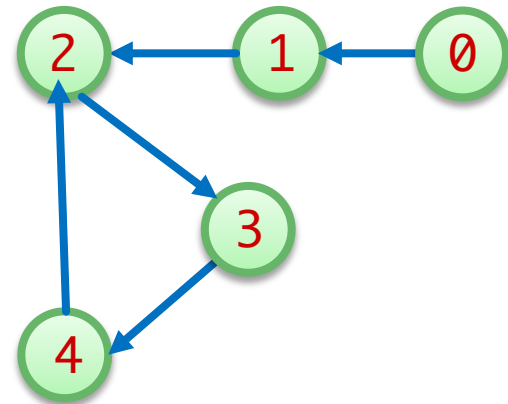
vértice  $\rightarrow$  # predecesores pendientes

1  $\rightarrow$  0

2  $\rightarrow$  2

3  $\rightarrow$  1

4  $\rightarrow$  1



Orden Topológico: 0

# Orden Topológico. DiGrafo cíclico

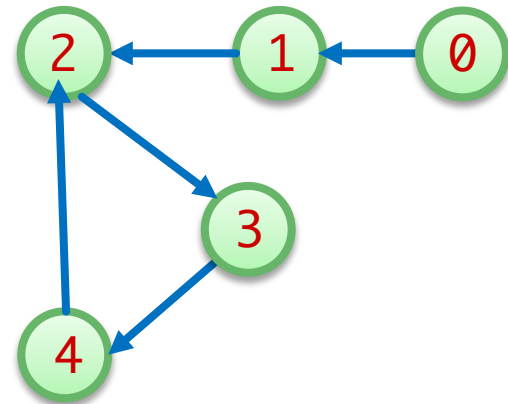
Eliminamos fuentes  $F = \{1\}$  del diccionario

**vértice -> # predecesores pendientes**

2 -> 2

3 -> 1

4 -> 1



Orden Topológico: 0

# Orden Topológico. DiGrafo cíclico

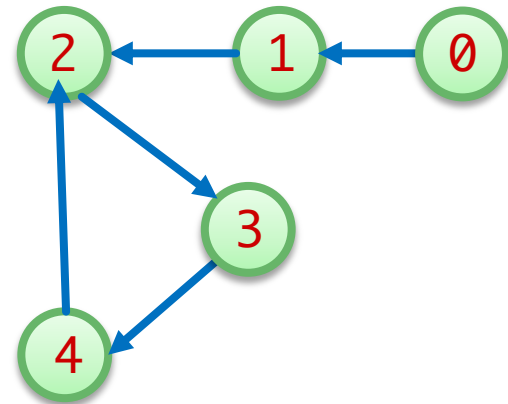
Añadimos fuentes  $F = \{1\}$  al orden topológico

**vértice -> # predecesores pendientes**

2 -> 2

3 -> 1

4 -> 1



Orden Topológico:  $0 < 1$

# Orden Topológico. DiGrafo cíclico

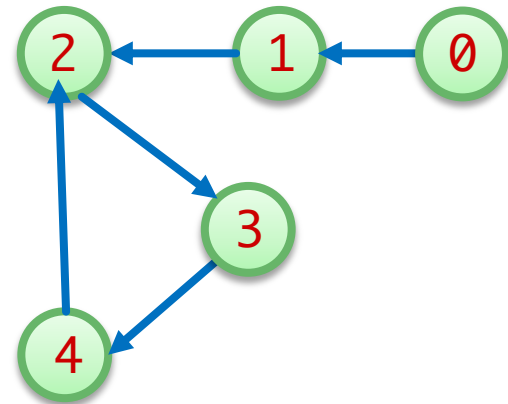
Restamos fuentes  $F = \{1\}$  del diccionario

**vértice -> # predecesores pendientes**

2 -> 1

3 -> 1

4 -> 1



Orden Topológico: 0 < 1

# Orden Topológico. DiGrafo cíclico

Seleccionamos fuentes  $F = \{\}$  del diccionario

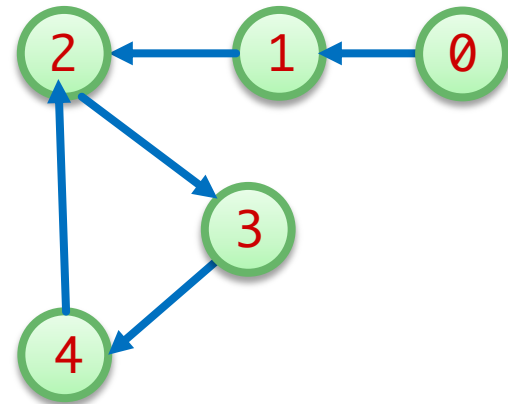
vértice  $\rightarrow$  # predecesores pendientes

2  $\rightarrow$  1

3  $\rightarrow$  1

4  $\rightarrow$  1

No hay fuentes. El  
DiGrafo tiene un ciclo y  
no existe  
un Orden Topológico



Orden Topológico: 0  $\prec$  1



# Orden Topológico y Paralelismo

- El orden topológico no guarda información sobre la posibilidad de realizar actividades en paralelo:

Orden Topológico: 8 < 0 < 2 < 7 < 1 < 3 < 6 < 4 < 5 < 9 < 10 < 11 < 12

- Recuperamos esta información almacenando el orden topológico en una colección anidada:

Orden Topológico: {8, 0} < {2, 7, 1} < {3, 6} < {4, 5} < {9} < {10, 11} < {12}

# Dos implementaciones alternativas

```
public class TopologicalSortingDic<V> {  
  
    private List<V> topSort;  
    private boolean hasCycle;  
  
    public TopologicalSortingDic(DiGraph<V> graph) {  
        // completar  
    }  
}  
  
public class TopologicalSortingDicPar<V> {  
  
    private List<Set<V>> topSort; // colección anidada  
    private boolean hasCycle;  
  
    public TopologicalSortingDicPar(DiGraph<V> graph) {  
        // completar  
    }  
}
```

# Salidas de TopSortDemo.java

## Para g1:

- ArrayList(3,7,5,8,11,2,9,10)
- ArrayList(HashSet(3,7),HashSet(5,8),HashSet(11),HashSet(2,9,10))

## Para g2:

- ArrayList(0,8,1,2,7,3,6,4,5,9,10,11,12)
- ArrayList(HashSet(0,8),HashSet(1,2,7),HashSet(3,6),HashSet(4,5),HashSet(9),HashSet(10,11),HashSet(12))

## Para g3:

- DiGraph is cyclic