

# **Test de Weisfeiler-Lehman**

Alexander Pinto - Marcelo Arenas

IIC3810

# Un poco de historia

- Test de isomorfismo de grafos basado en etiquetado canónico.
- Weisfeiler & Lehman (**WL**), 1968, algoritmo de etiquetado de vértices (**1-WL**, **2-WL**).
- Babai & Mathon, 1979, algoritmo de etiquetado de tuplas de vértices ( **$k$ -WL**).
- Cai, Furer & Immerman, 1992, variante del algoritmo ( **$k$ -folklore-WL**).

# Equivalencia por color

El algoritmo define una relación de equivalencia en el color (etiqueta) de cada nodo de un grafo.

- $C$  es una función que devuelve el color de un nodo.
- $\sim_C$  es una relación de equivalencia tal que  $u \sim_C v$  si y sólo si  $C(u) = C(v)$ .

# Refinamiento

Dadas: relaciones de equivalencia  $R$  y  $S$

- $S$  es un refinamiento de  $R$  si toda clase de equivalencia de  $S$  es un subconjunto de una clase de equivalencia de  $R$ .
- Además,  $S$  es un refinamiento estricto de  $R$  si al menos una clase de equivalencia de  $S$  es un subconjunto propio de una clase de equivalencia de  $R$ .

# Algoritmo 1-WL

**Input:**  $G = (V, E)$

1. **for each**  $v \in V$  **do**

2.    $C[v] \leftarrow 1$

3. **repeat**

4.    $C^{old} \leftarrow C$

5. **for each**  $v \in V$  **do**

6.      $C[v] \leftarrow \text{hash}\left(C^{old}[v], \{\{C^{old}[w] \mid w \in N_G(v)\}\}\right)$

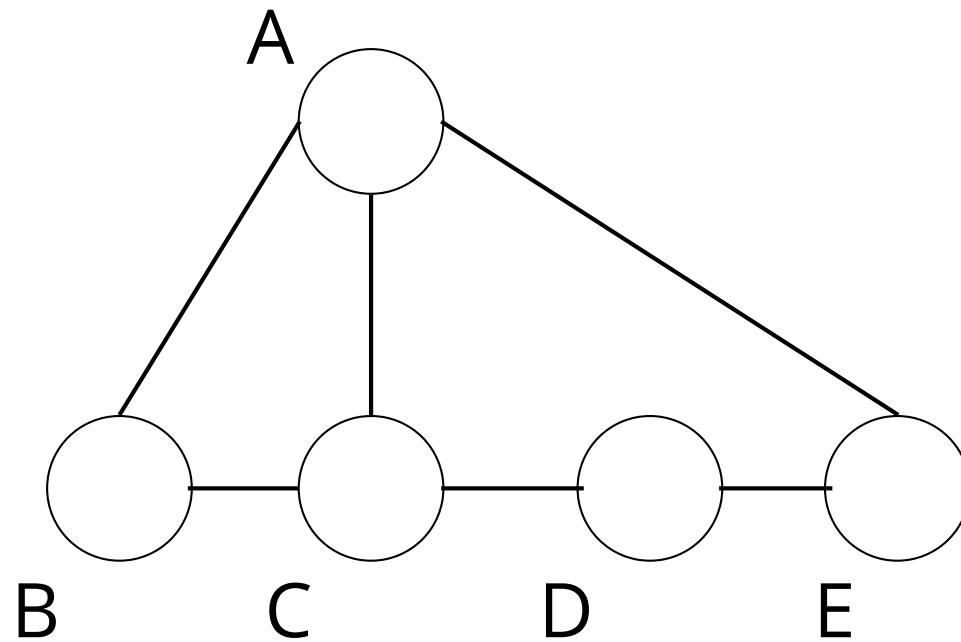
$$N_G(v) = \{u \in V \mid \{u, v\} \in E\}$$



7. **until not**  $\sim_C$  es un refinamiento estricto de  $\sim_{C^{old}}$

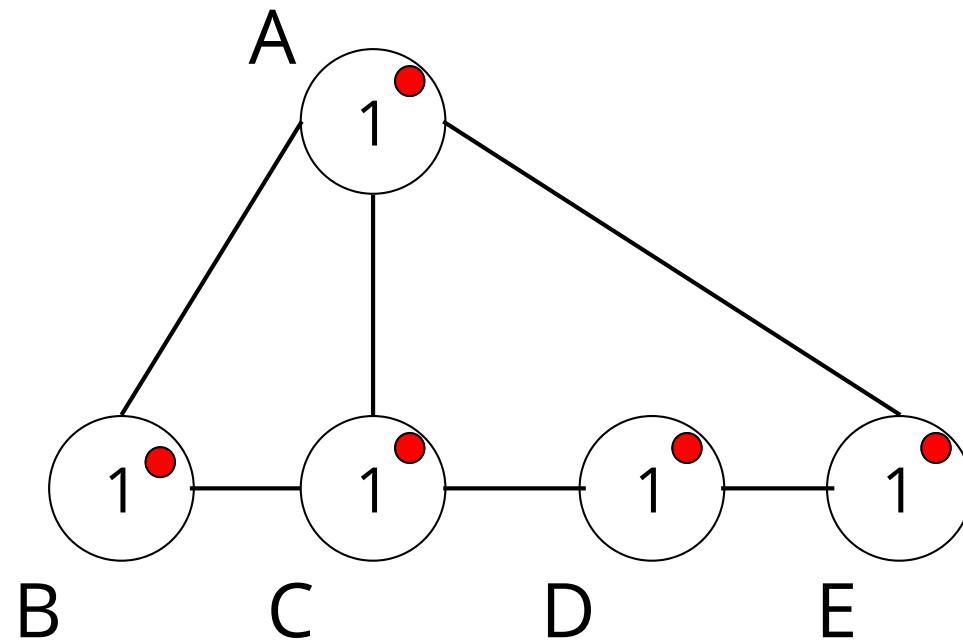
8. **return**  $C$

# Algoritmo 1-WL

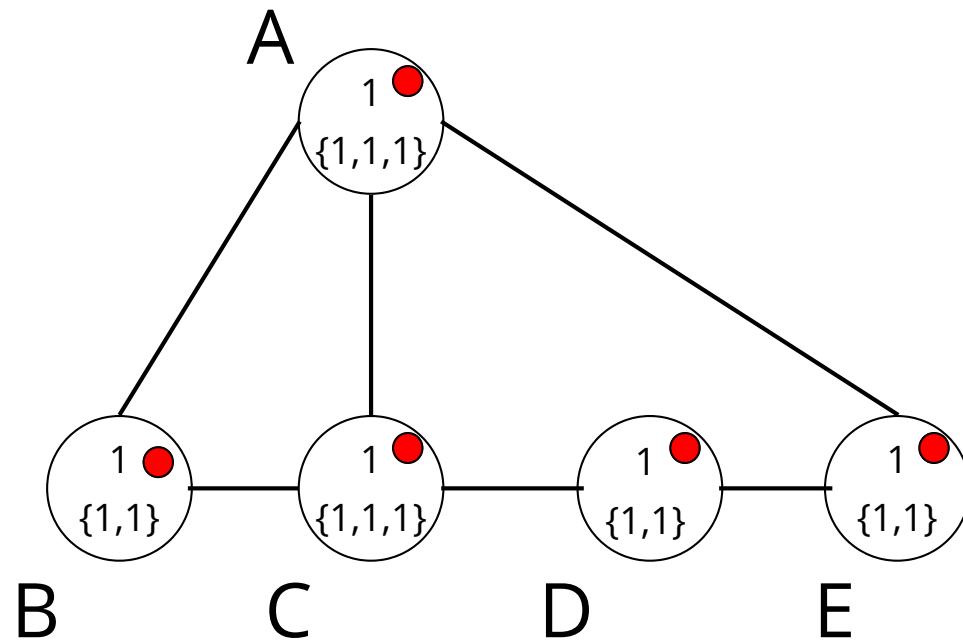


1. **for each**  $v \in V$  **do**

2.    $C[v] \leftarrow 1$

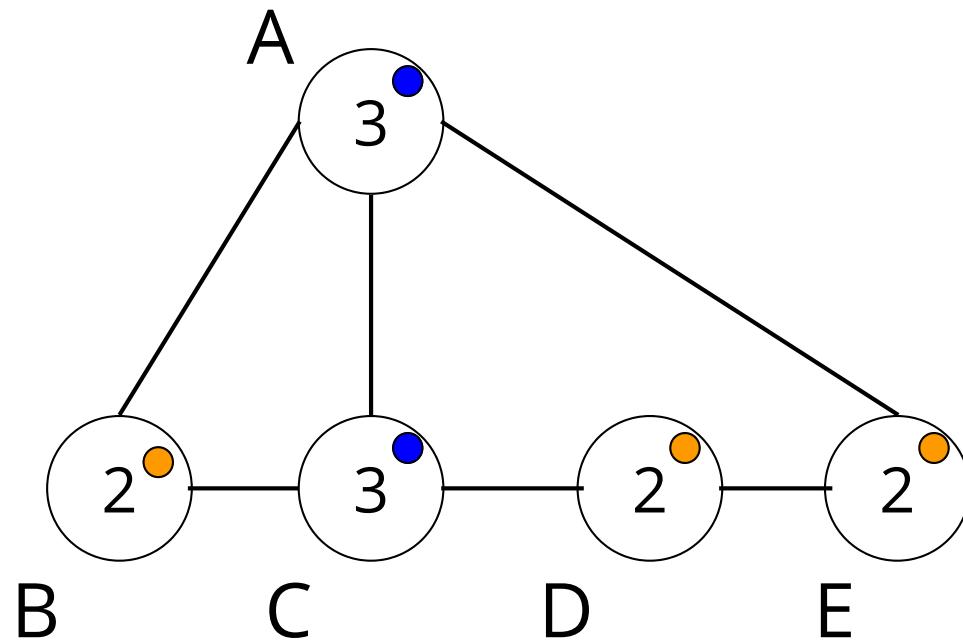


$$\begin{array}{c} 1^{\bullet} \\ \quad : \quad C^{old} \\ \{1, 1, 1\} \quad : \quad \{\{C^{old}[w] \mid w \in N_G(v)\}\} \end{array}$$

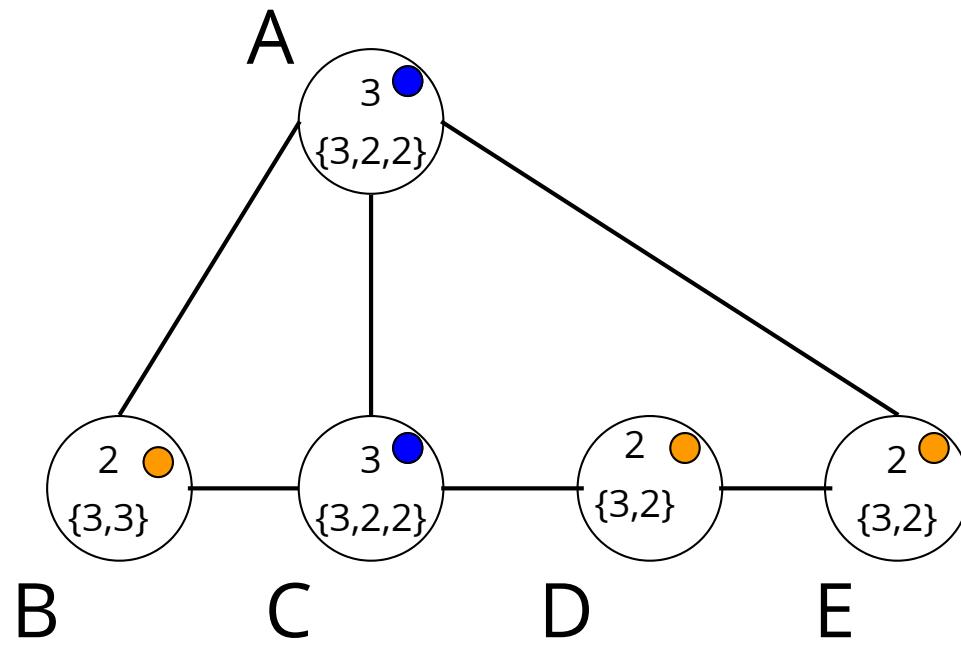


$$C[v] \leftarrow \text{hash}(C^{old}[v], \{[C^{old}[w] \mid w \in N_G(v)]\})$$

$$\begin{array}{ccc} 3^{\bullet} & \xleftarrow{\hspace{1cm}} & 1^{\bullet} \\ & & : \quad C^{old} \\ \{1, 1, 1\} & : & \{[C^{old}[w] \mid w \in N_G(v)]\} \end{array}$$

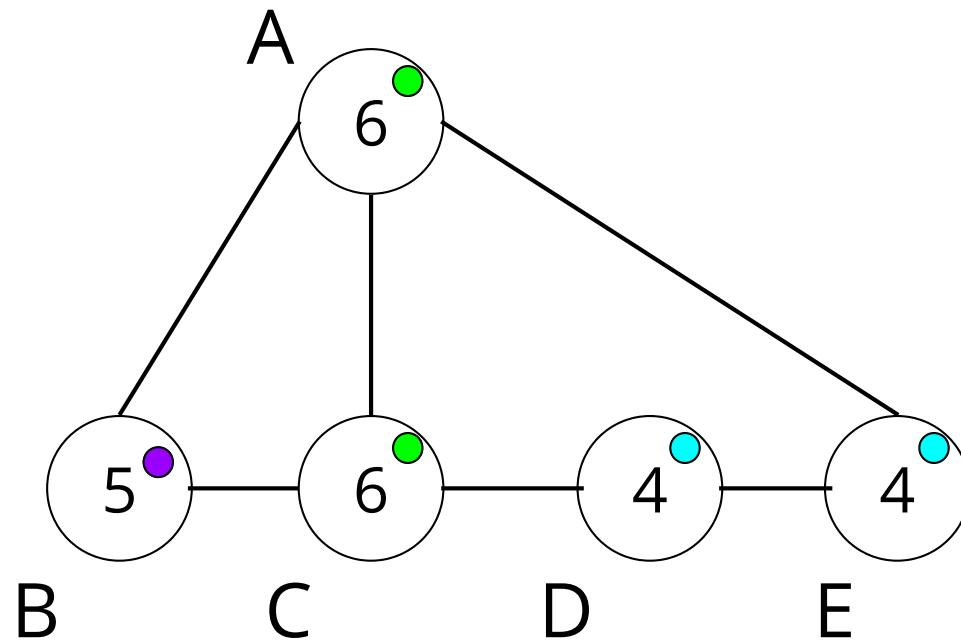


$$\begin{array}{lcl} 3^{\bullet} & : & C^{old} \\ \{3, 2, 2\} & : & \{\{C^{old}[w] \mid w \in N_G(v)\}\} \end{array}$$



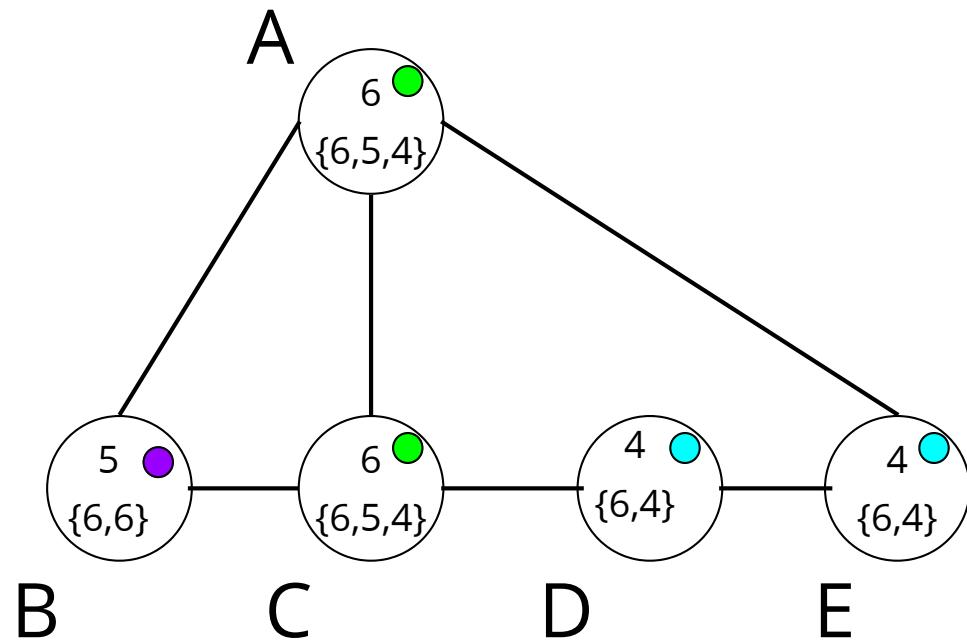
$$C[v] \leftarrow \text{hash}(C^{old}[v], \{[C^{old}[w] \mid w \in N_G(v)]\})$$

$$\begin{array}{ccc} 6^{\bullet} & \xleftarrow{\hspace{1cm}} & 3^{\bullet} \\ & & : \quad C^{old} \\ \{3, 2, 2\} & : & \{[C^{old}[w] \mid w \in N_G(v)]\} \end{array}$$

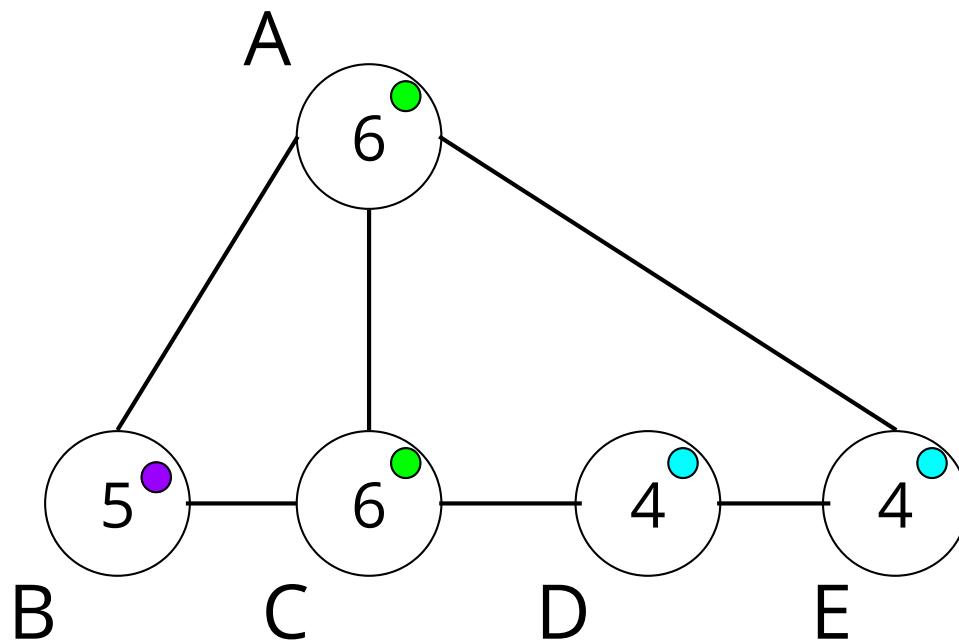


$6^*$  :  $C^{old}$

$\{6, 5, 4\}$  :  $\{C^{old}[w] \mid w \in N_G(v)\}$



$$C[v] \leftarrow \text{hash}(C^{old}[v], \{C^{old}[w] \mid w \in N_G(v)\})$$



8. **until not**  $\sim_C$  es un refinamiento estricto de  $\sim_{C^{old}}$
9. **return**  $C$

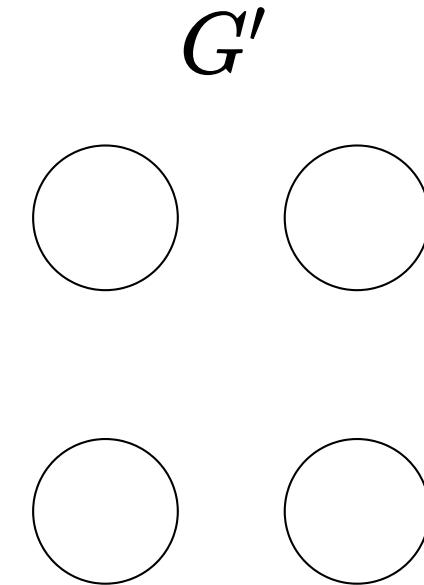
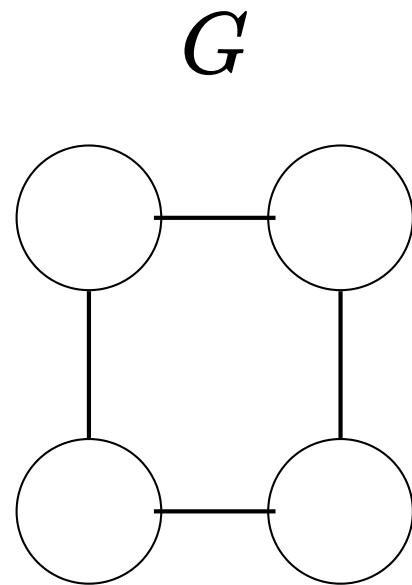
$$C = \{ A \rightarrow 6, B \rightarrow 5, C \rightarrow 6, D \rightarrow 4, E \rightarrow 4 \}$$

# Test de isomorfismo WL

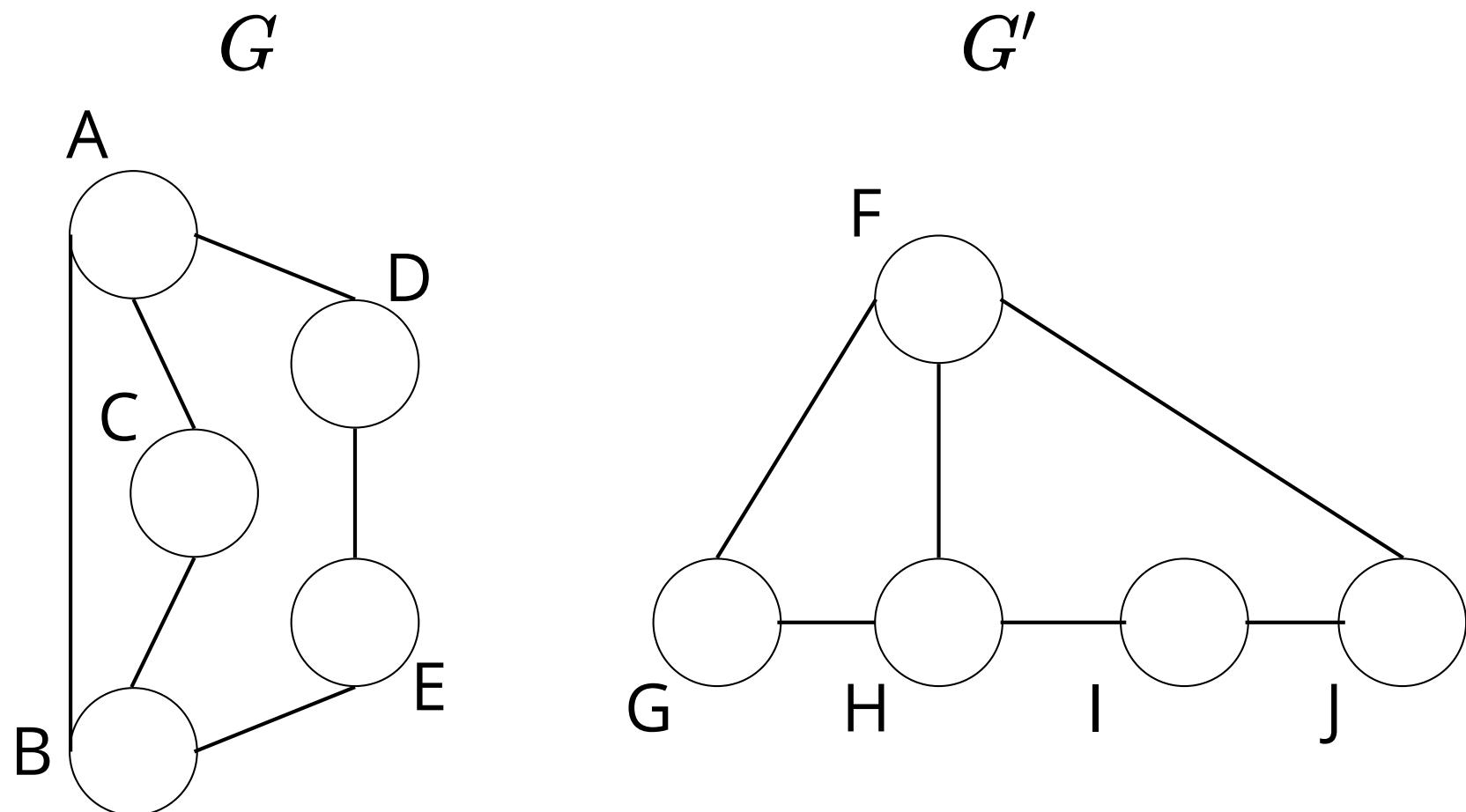
**Input:**  $G = (V, E)$  y  $G' = (V', E')$

1. **if**  $|V| \neq |V'|$  **or**  $|E| \neq |E'|$  **then**  
**return**  $G$  no es isomorfo a  $G'$
2.  $H \leftarrow G \uplus G'$
3.  $C = \text{1-WL}(H)$
4. Ordenar  $C(G)$  y  $C(G')$  según color
5. **for**  $i = 1$  **to**  $|V|$  **do**
6.   **if**  $C(G)[i] \neq C(G')[i]$  **then**  
**return**  $G$  no es isomorfo a  $G'$
8. **return**  $G$  es isomorfo a  $G'$

¿Por qué colocamos a  $G$  y  $G'$  en el mismo grafo  $H$ ?



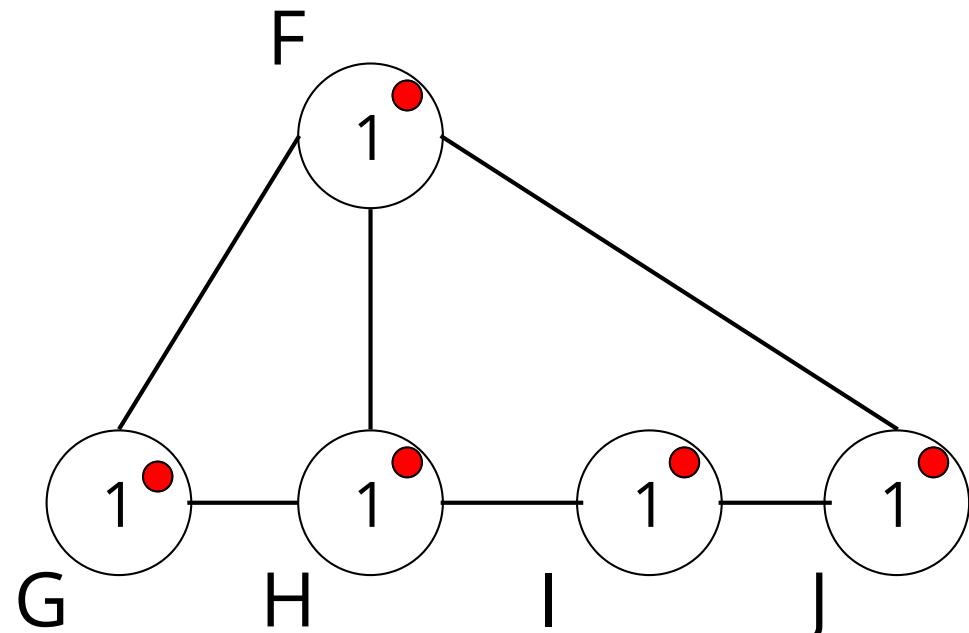
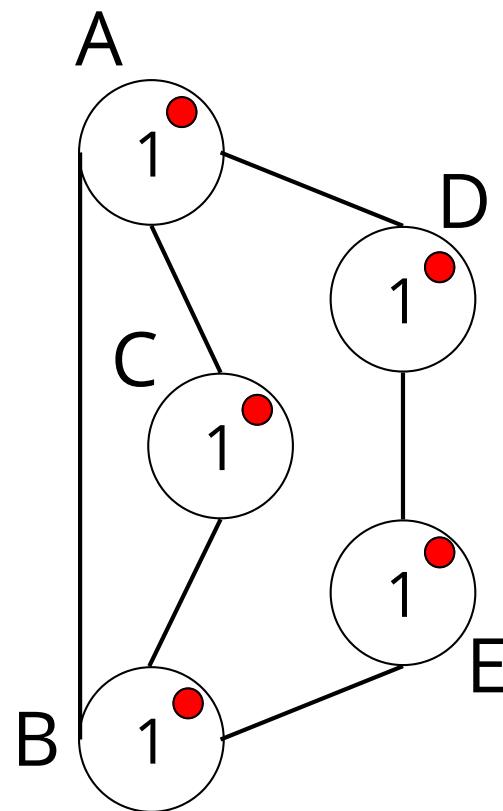
# Test de isomorfismo WL



# Algoritmo 1-WL

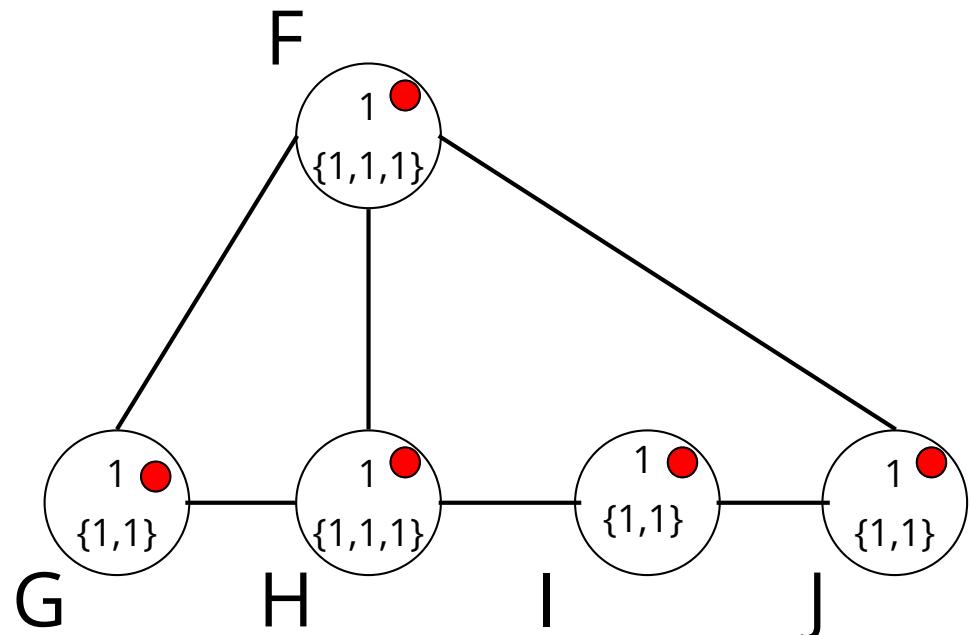
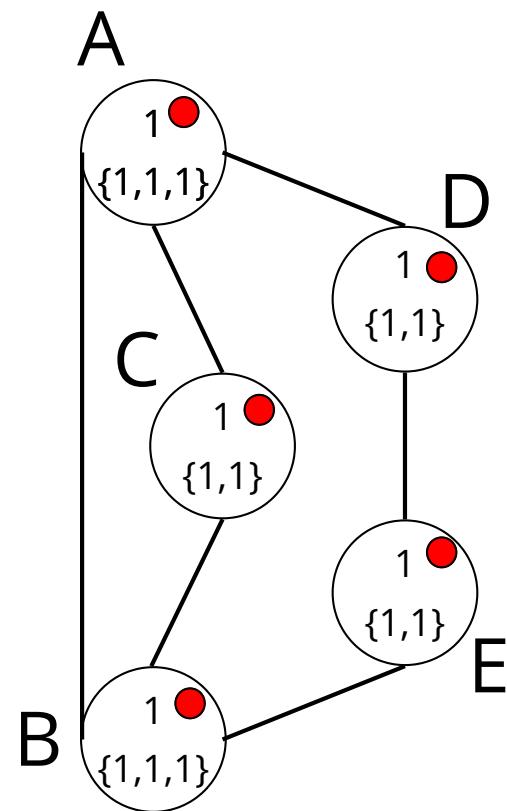
1. **for each**  $v \in V$  **do**

2.    $C[v] \leftarrow 1$



$1^{\bullet}$  :  $C^{old}$

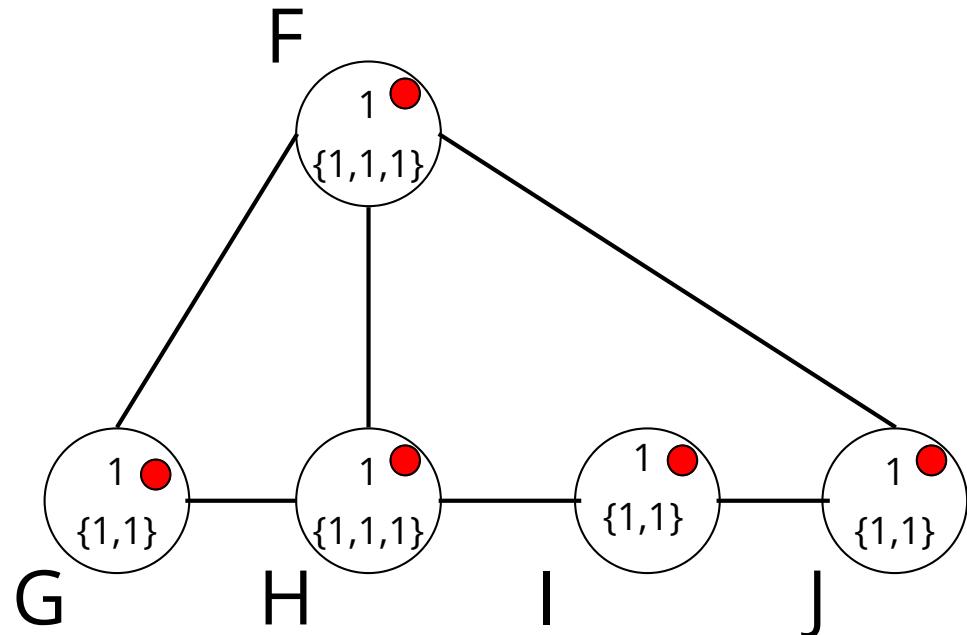
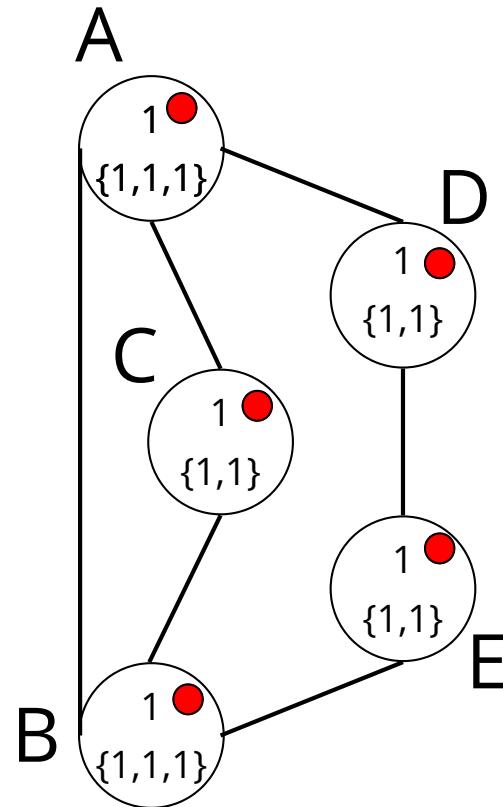
$\{1, 1, 1\}$  :  $\{C^{old}[w] \mid w \in N_H(v)\}$



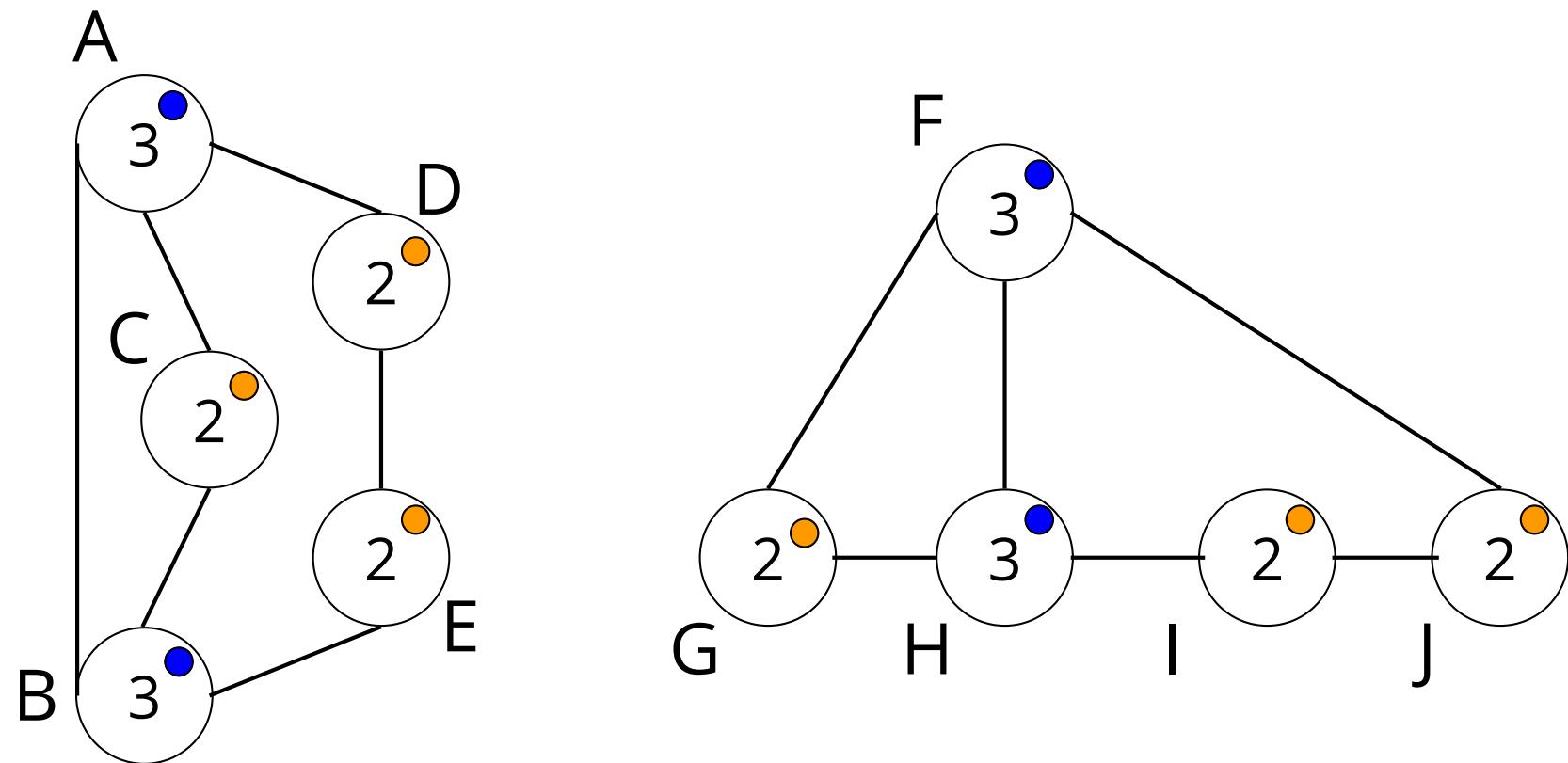
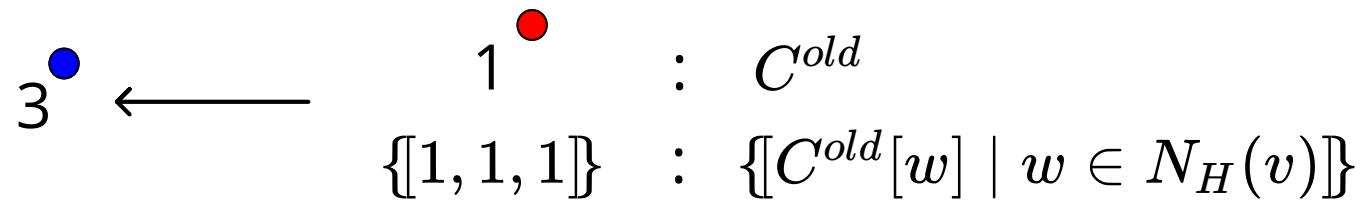
$$C[v] \leftarrow \text{hash}(C^{old}[v], \{C^{old}[w] \mid w \in N_H(v)\})$$

1  :  $C^{old}$

$\{1, 1, 1\}$  :  $\{C^{old}[w] \mid w \in N_H(v)\}$

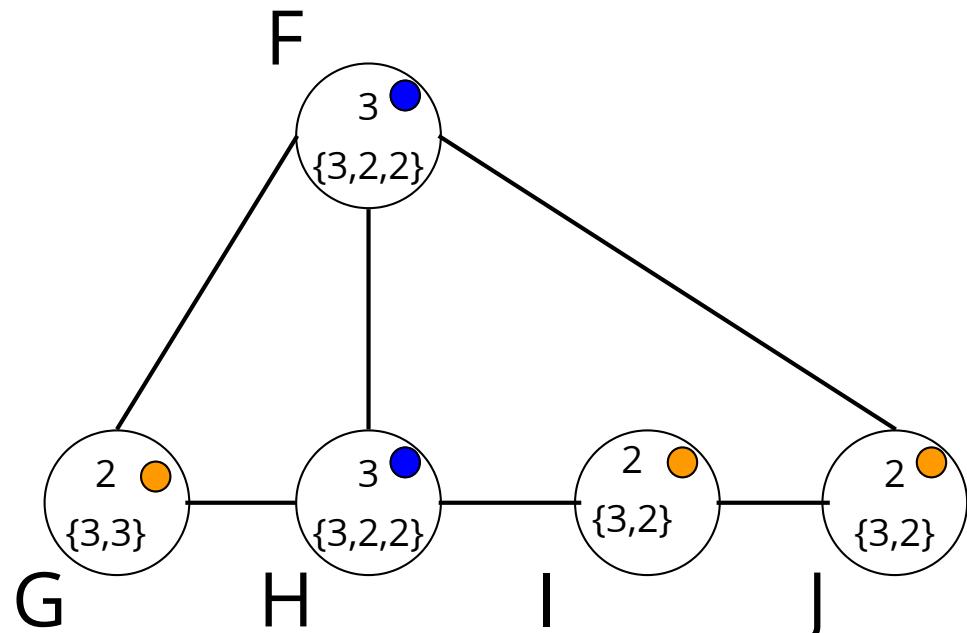
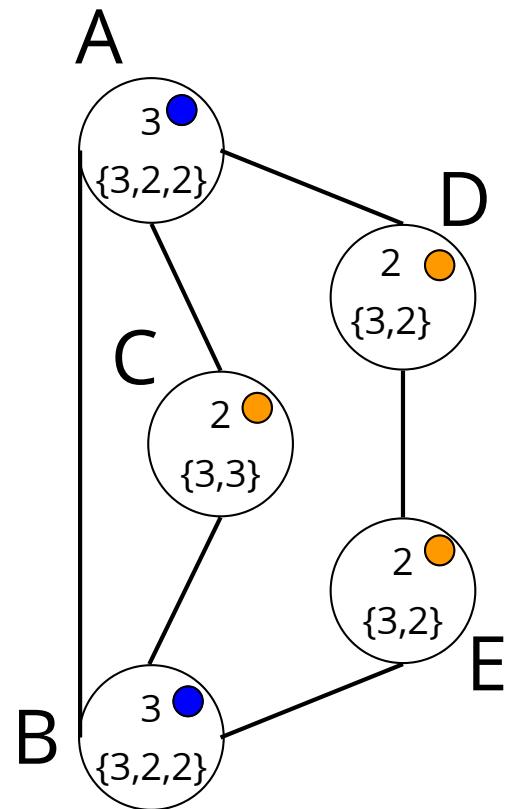


$$C[v] \leftarrow \text{hash}(C^{old}[v], \{C^{old}[w] \mid w \in N_H(v)\})$$



$3^*$  :  $C^{old}$

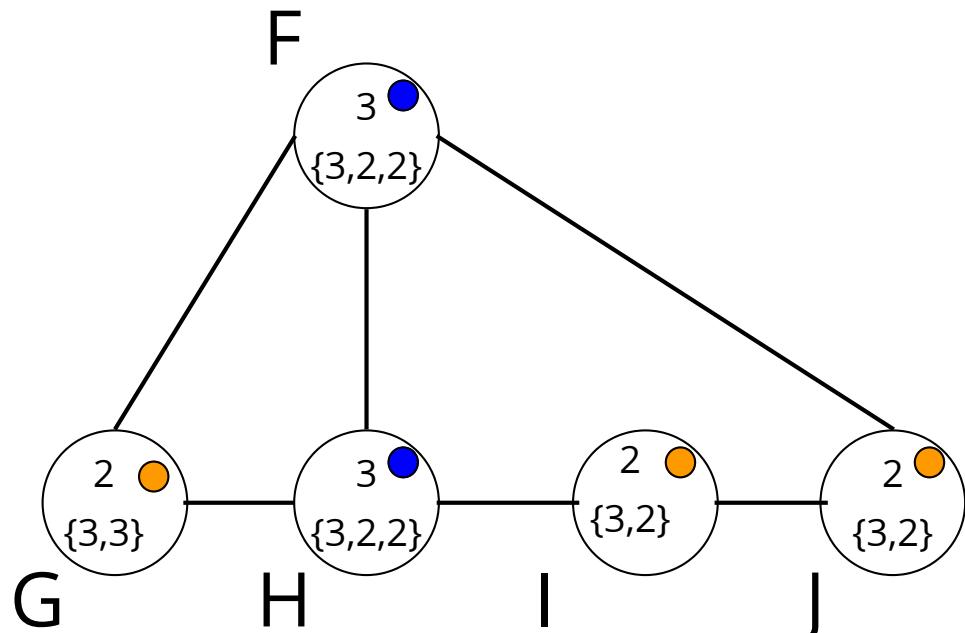
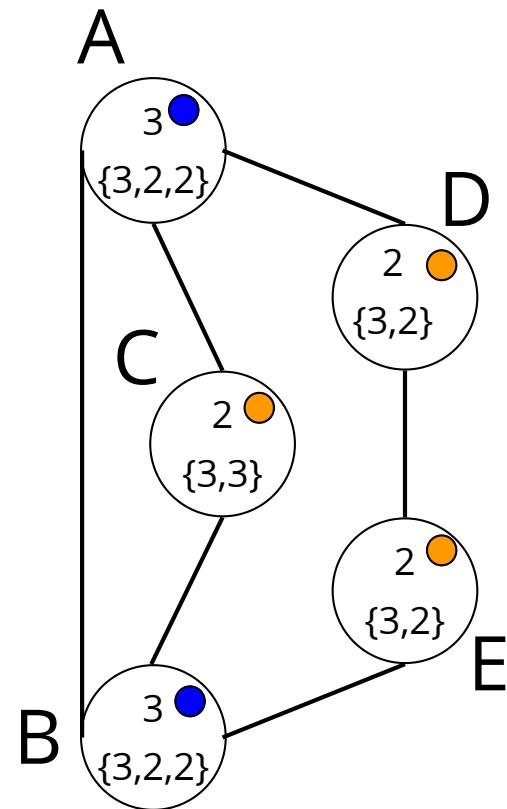
$\{\{3, 2, 2\}\}$  :  $\{\{C^{old}[w] \mid w \in N_H(v)\}\}$



$$C[v] \leftarrow \text{hash}(C^{old}[v], \{[C^{old}[w] \mid w \in N_H(v)]\})$$

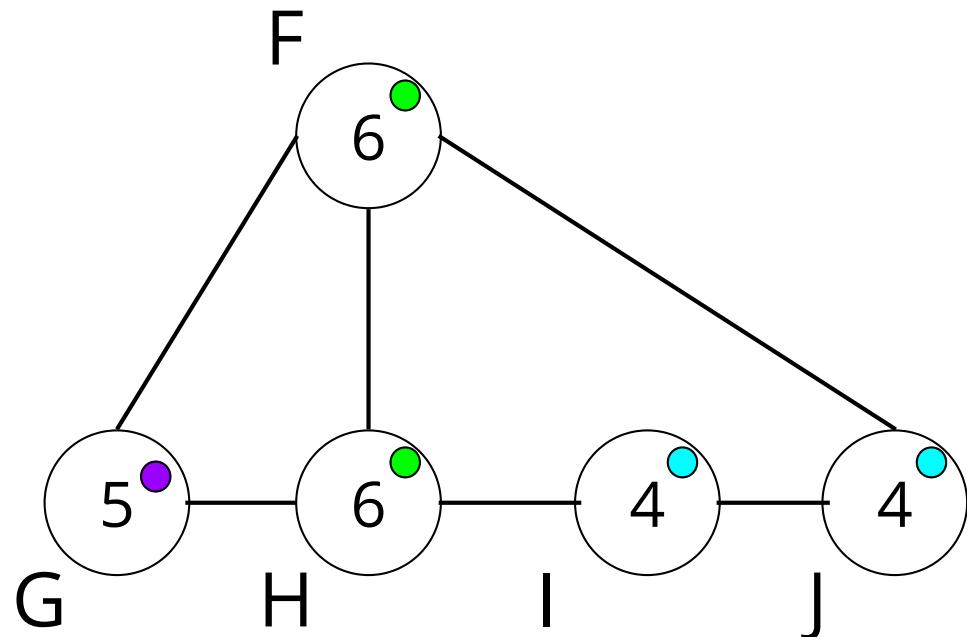
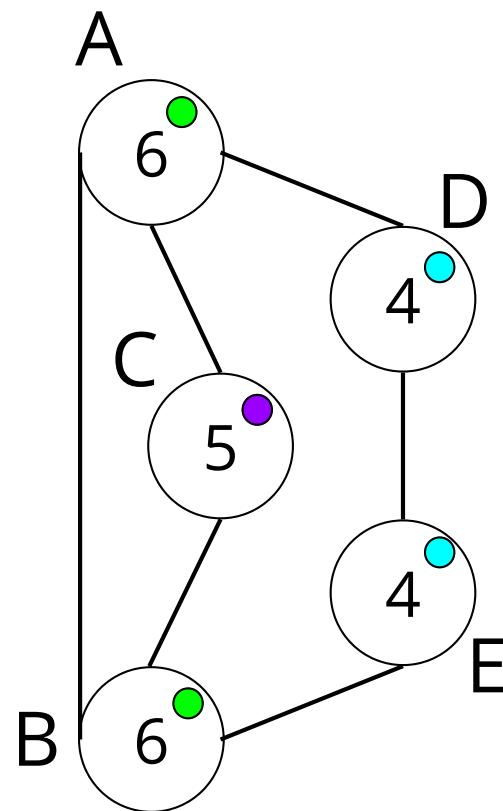
$3^*$  :  $C^{old}$

$\{3, 2, 2\}$  :  $\{[C^{old}[w] \mid w \in N_H(v)]\}$



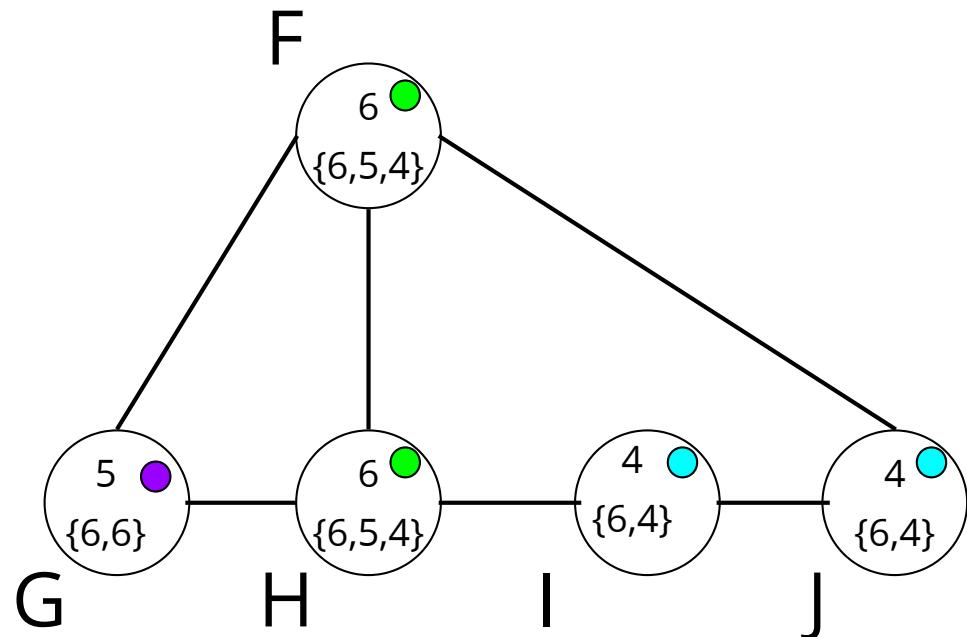
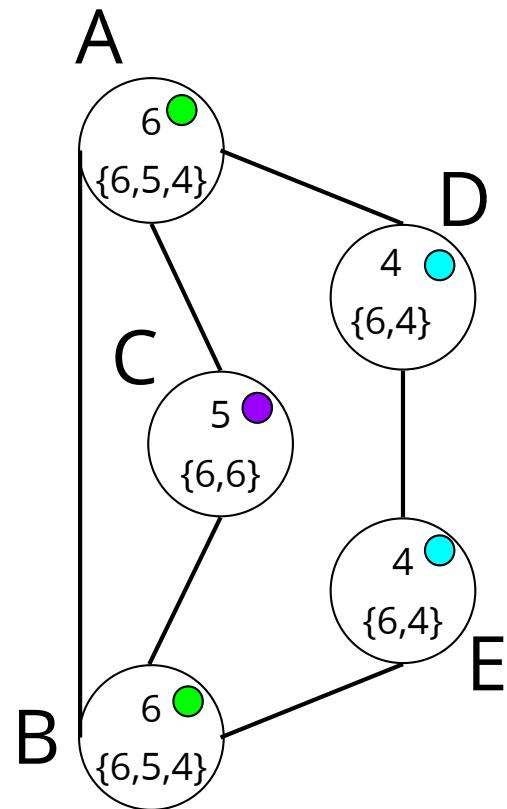
$$C[v] \leftarrow \text{hash}(C^{old}[v], \{C^{old}[w] \mid w \in N_H(v)\})$$

$$\begin{array}{ccc} 6^{\bullet} & \xleftarrow{\hspace{1cm}} & 3^{\bullet} \\ \{3, 2, 2\} & : & \{C^{old}[w] \mid w \in N_H(v)\} \end{array}$$

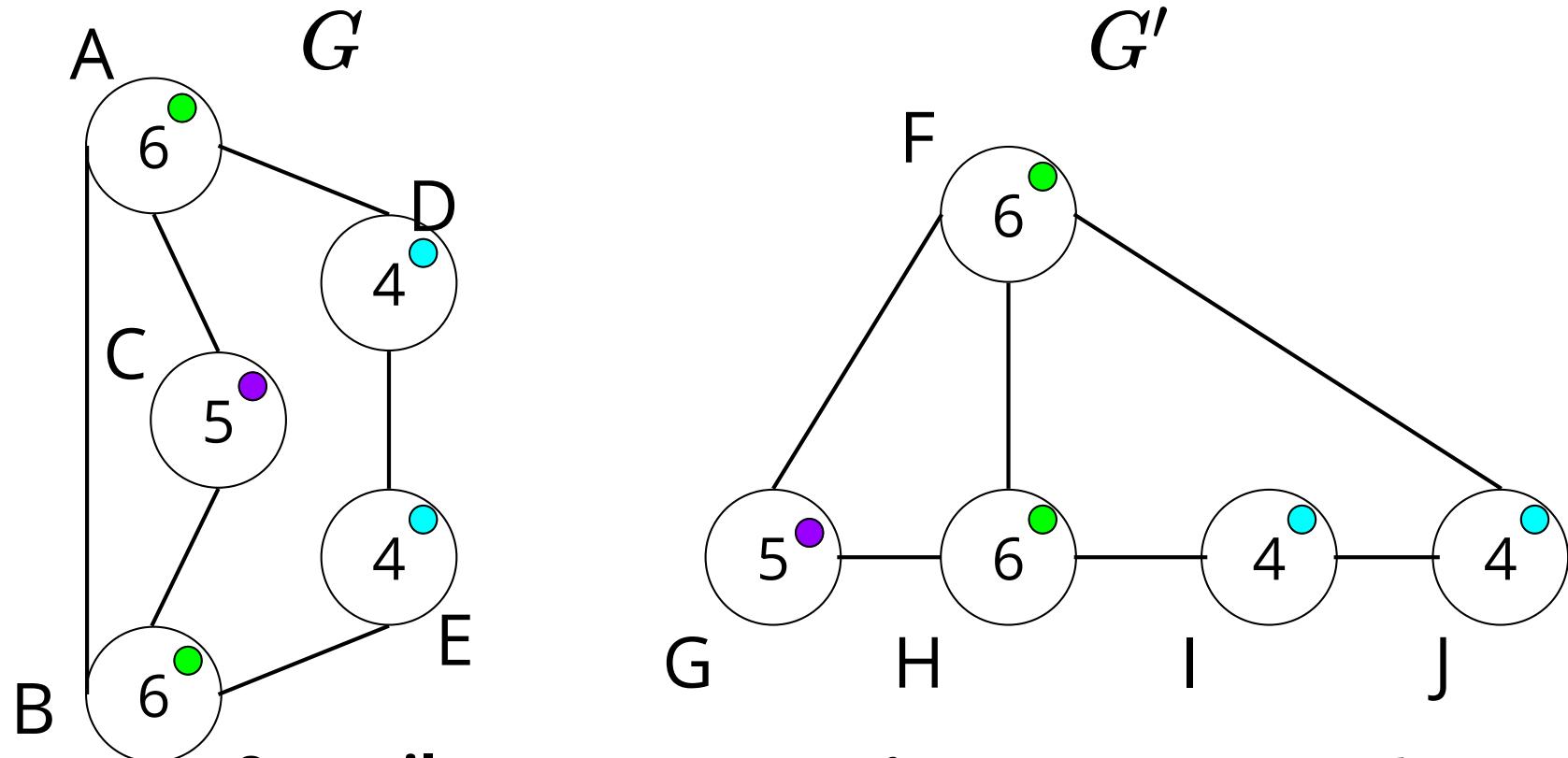


$6^{\bullet}$  :  $C^{old}$

$\{[6, 5, 4]\}$  :  $\{C^{old}[w] \mid w \in N_H(v)\}$



$$C[v] \leftarrow \text{hash}(C^{old}[v], \{C^{old}[w] \mid w \in N_H(v)\})$$



8. **until not**  $\sim_C$  es un refinamiento estricto de  $\sim_{C^{old}}$

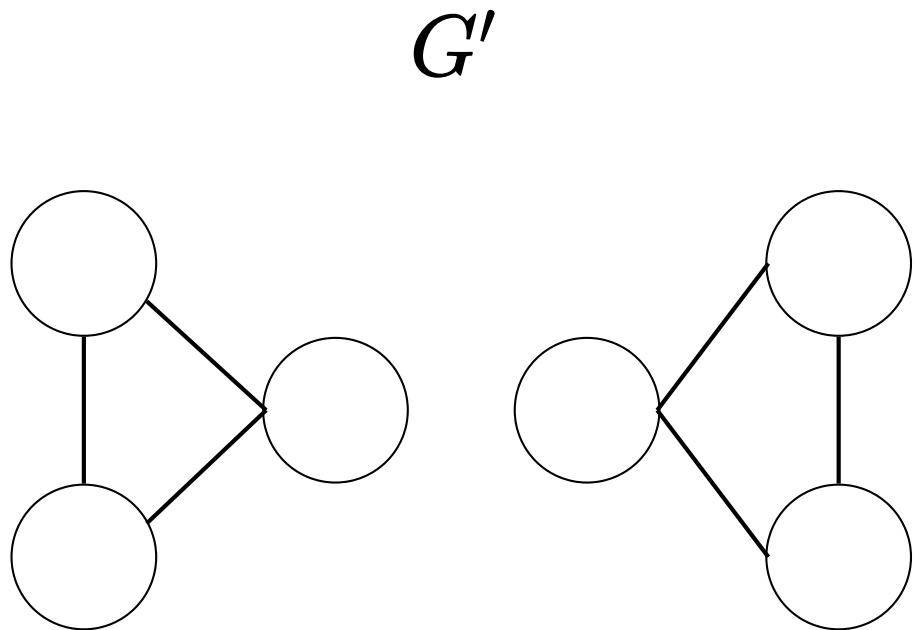
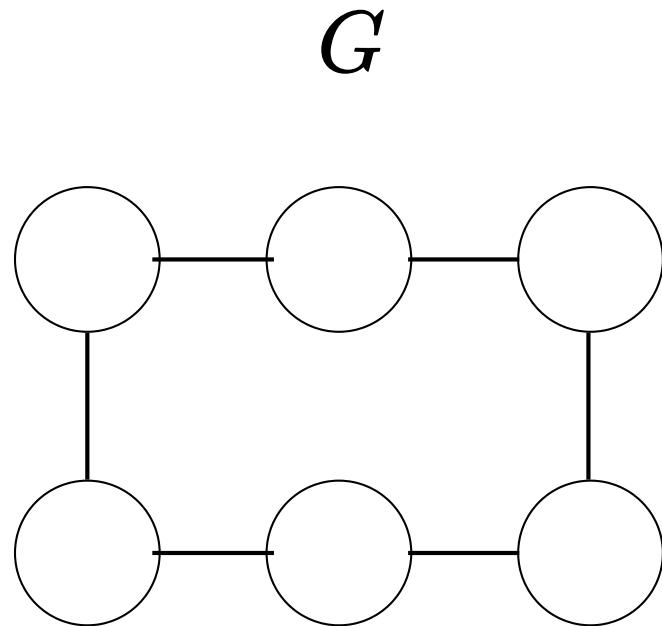
9. **return**  $C$

$$C(G) = \{ D \rightarrow 4, E \rightarrow 4, C \rightarrow 5, A \rightarrow 6, B \rightarrow 6 \}$$

$$C(G') = \{ I \rightarrow 4, J \rightarrow 4, G \rightarrow 5, F \rightarrow 6, H \rightarrow 6 \}$$

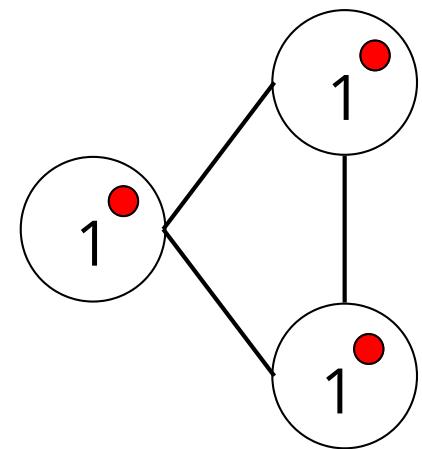
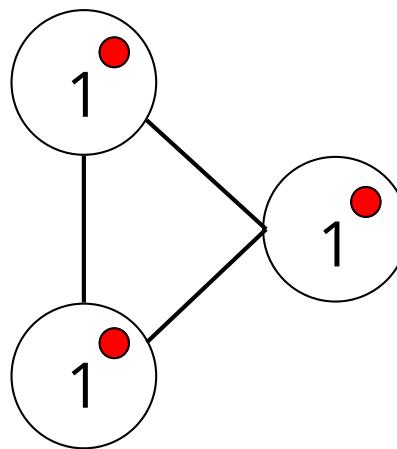
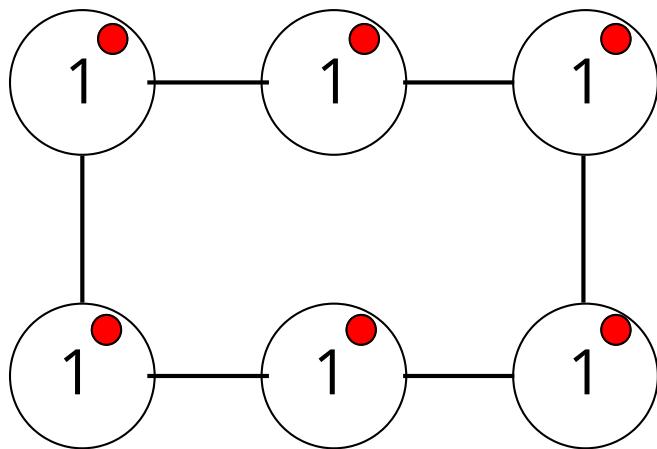


# Test de isomorfismo WL



# Algoritmo 1-WL

1. **for each**  $v \in V$  **do**
2.    $C[v] \leftarrow 1$

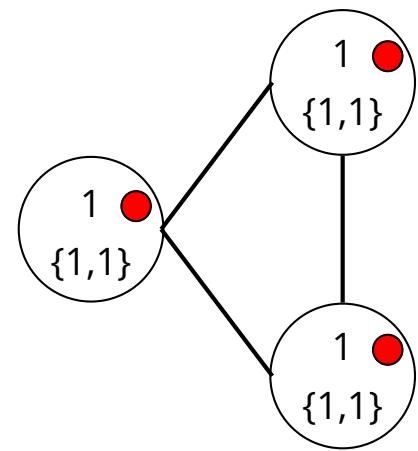
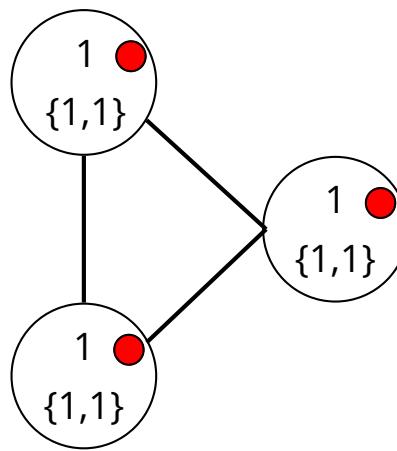
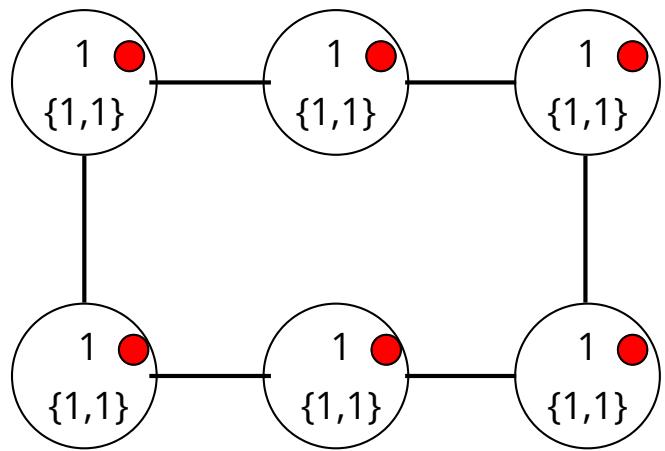


1

:  $C^{old}$

{1, 1}

:  $\{C^{old}[w] \mid w \in N_H(v)\}$



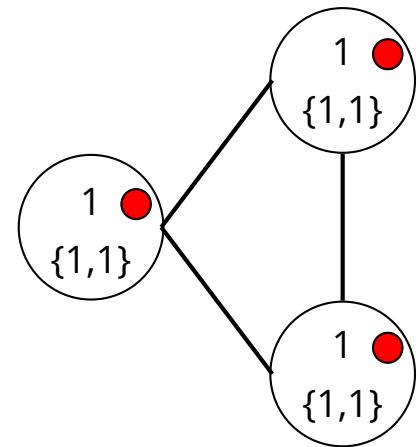
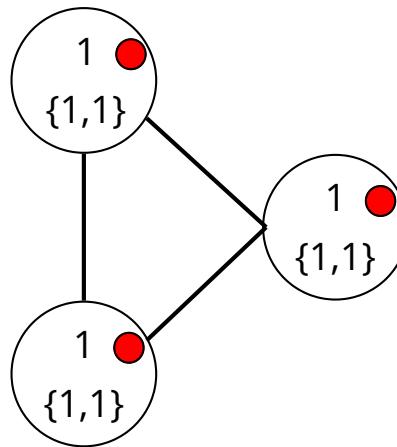
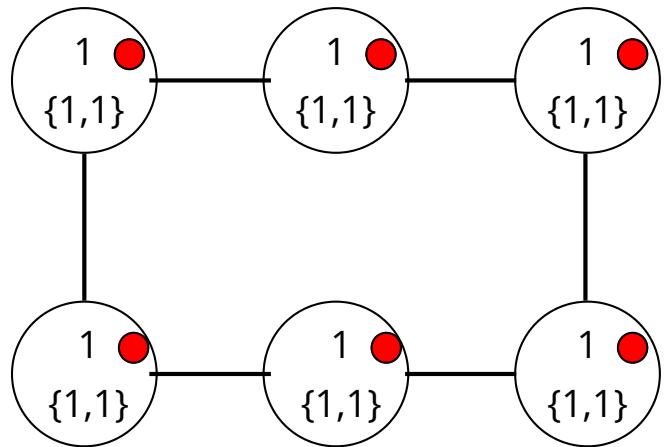
$$C[v] \leftarrow \text{hash}(C^{old}[v], \{[C^{old}[w] \mid w \in N_H(v)]\})$$

1

:  $C^{old}$

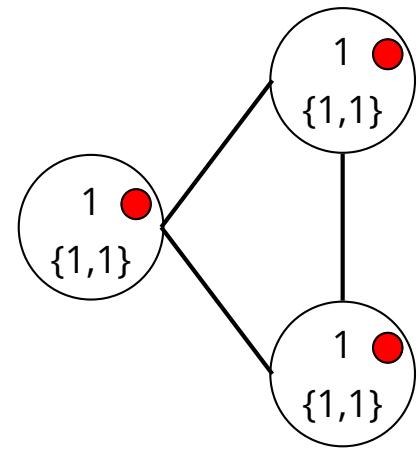
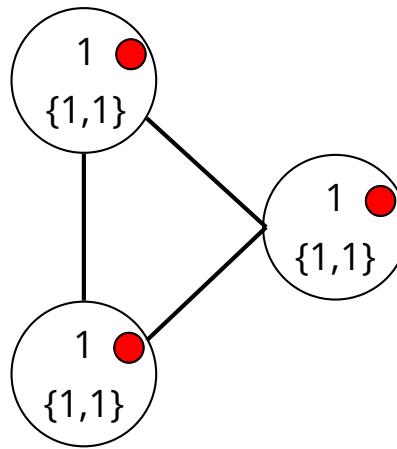
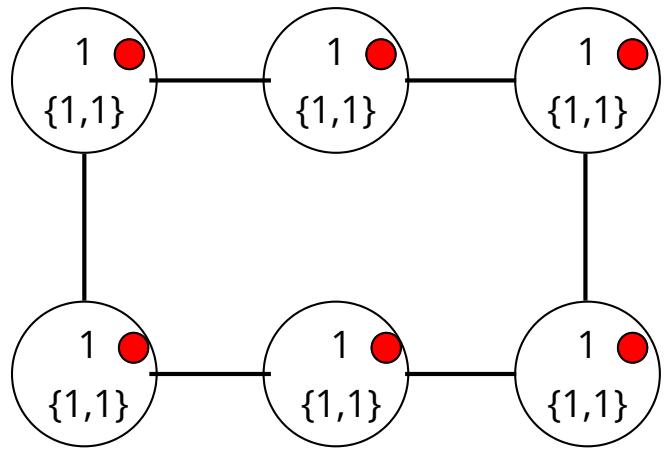
{1, 1}

:  $\{[C^{old}[w] \mid w \in N_H(v)]\}$



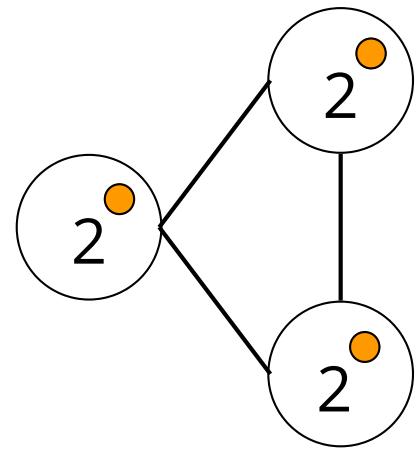
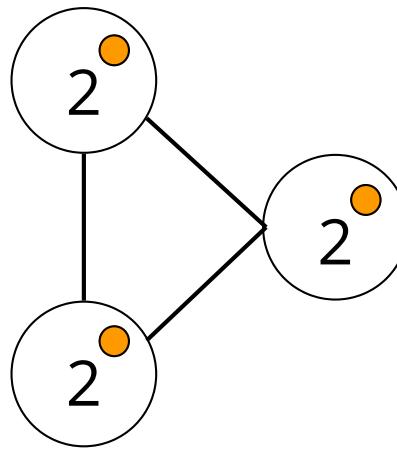
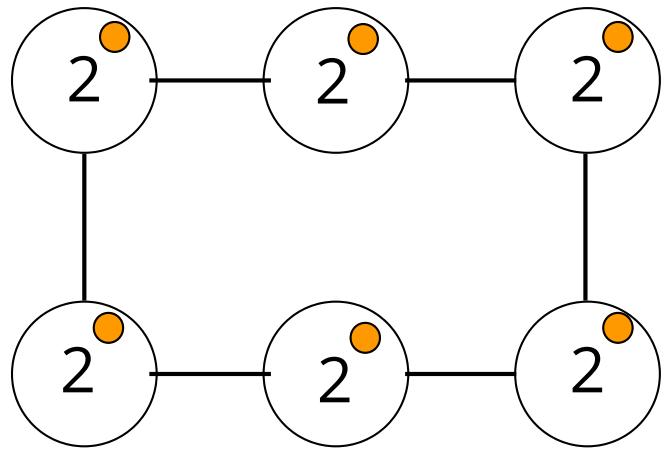
$$C[v] \leftarrow \text{hash}(C^{old}[v], \{C^{old}[w] \mid w \in N_H(v)\})$$

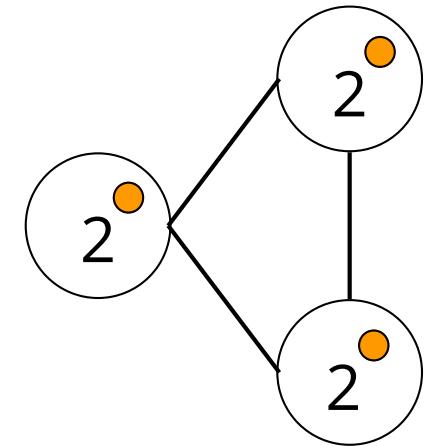
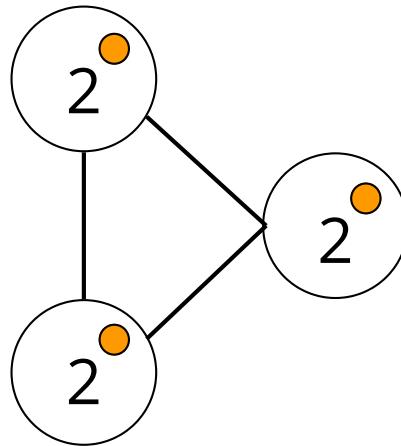
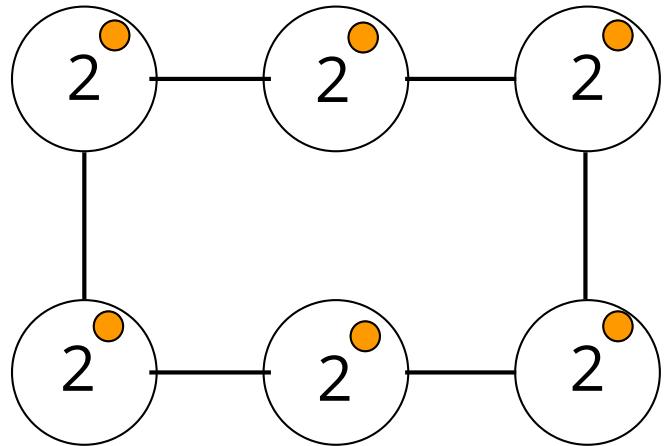
$$\begin{array}{ccc} 2^{\bullet} & \xleftarrow{\hspace{1cm}} & 1^{\bullet} \\ : & & : \\ \{1, 1\} & & \{C^{old}[w] \mid w \in N_H(v)\} \end{array}$$



$$C[v] \leftarrow \text{hash}(C^{old}[v], \{[C^{old}[w] \mid w \in N_H(v)]\})$$

$$\begin{array}{ccc} 2^{\bullet} & \xleftarrow{\hspace{1cm}} & 1^{\bullet} \\ \{1, 1\} & : & C^{old} \\ & : & \{[C^{old}[w] \mid w \in N_H(v)]\} \end{array}$$





8. **until**  $\text{not } \sim_C$  es un refinamiento estricto de  $\sim_{C^{old}}$

9. **return**  $C$

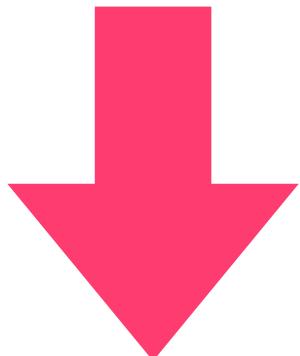
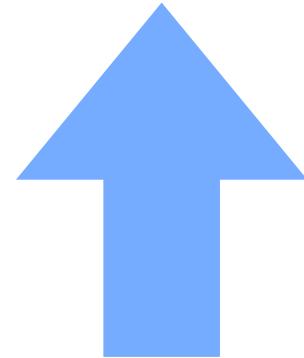
$$C(G) = \{2, 2, 2, 2, 2, 2\}$$

$$C(G') = \{2, 2, 2, 2, 2, 2\}$$



## Pros de 1-WL

- Simple y fácil de implementar
- Produce formas normales para todas los grafos de  $n$ -vértices excepto una fracción  $n^{-1/7}$



## Cons de 1-WL

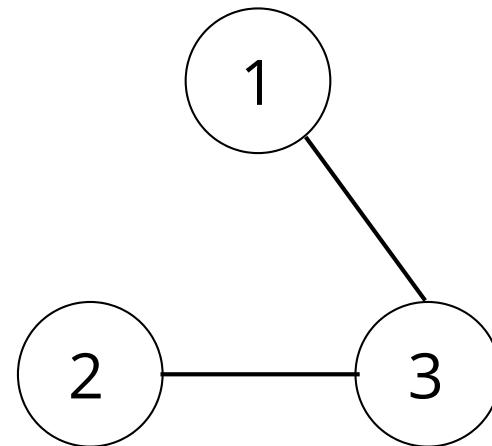
- Existe clases de grafos para los cuales el algoritmo es inútil, por ejemplo grafos regulares (el mismo grado en cada vértice)

# Algoritmo $k$ -WL

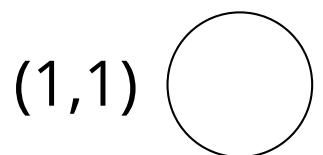
# Coloración de tuplas

Dado un grafo  $G = (V, E)$  y tuplas  $\bar{u} = (u_1, \dots, u_k)$  y  $\bar{v} = (v_1, \dots, v_k)$  en  $V^k$ .

$\bar{u}$  y  $\bar{v}$  son pintadas del mismo color si la función  $f(u_i) = v_i$  es un isomorfismo entre los subgrafos de  $G$  inducidos por  $\{u_1, \dots, u_k\}$  y  $\{v_1, \dots, v_k\}$ .



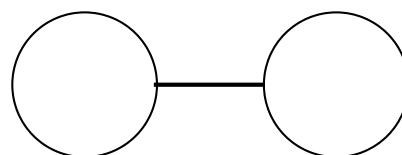
(1,1)



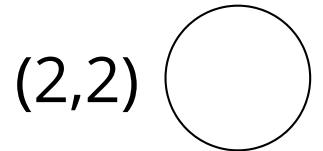
(1,1)



(1,3)



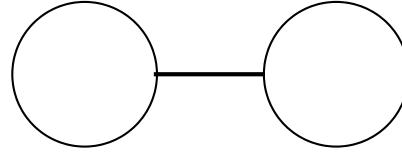
(2,2)



(2,2)



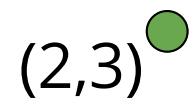
(2,3)



(1,3)



(2,3)

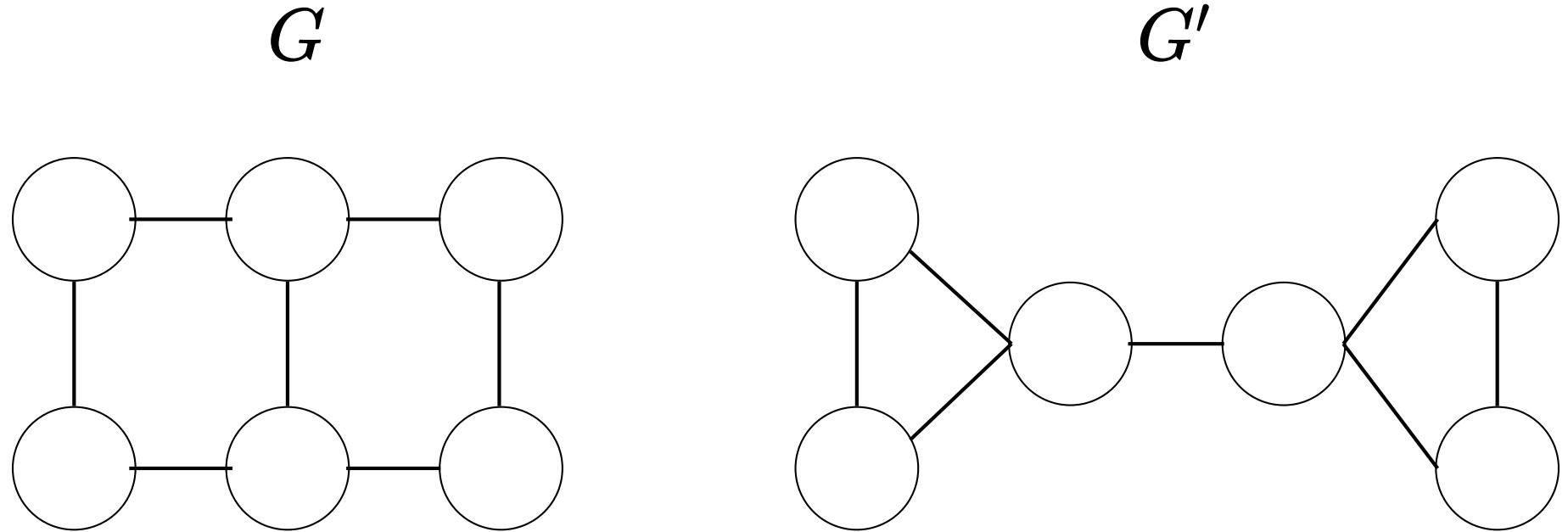


# Algoritmo 2-WL

**Input:**  $G = (V, E)$

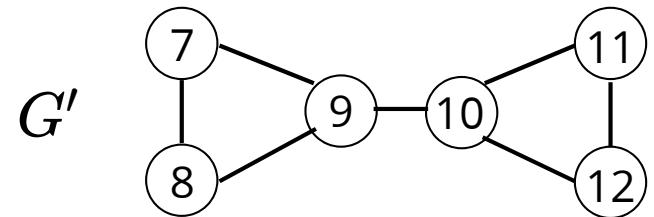
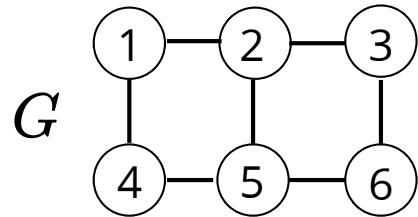
1.  $C \leftarrow \text{ColoreoInicial}(G)$
2. **repeat**
3.    $C^{old} \leftarrow C$
4.   **for each**  $(u, v) \in V^2$  **do**
5.      $\chi[1] \leftarrow \{C^{old}[(n, v)] \mid n \in V\}$
6.      $\chi[2] \leftarrow \{C^{old}[(u, n)] \mid n \in V\}$
7.      $C[(u, v)] \leftarrow \text{hash}\left(C^{old}[(u, v)], \chi[1], \chi[2]\right)$
8. **until not**  $\sim_C$  es un refinamiento estricto de  $\sim_{C^{old}}$
9. **return**  $C$

# Test de isomorfismo 2-WL



¿Qué resultado da 1-WL sobre estos grafos?

## ColoreoInicial( $G$ )



	1	2	3	4	5	6
1	C	B	Z	B	Z	Z
2		C	B	Z	B	Z
3	B		C	Z	Z	B
4		B		C	B	Z
5	B		B		C	B
6	B		B			C

	7	8	9	10	11	12
7	C	B	B	Z	Z	Z
8		C	B	Z	Z	Z
9			C	B	Z	Z
10	B	B		C	B	B
11	B	B	B		C	B
12	B	B	B			C

1	C	B	Z	B	Z	Z
2		C	B	Z	B	Z
3			C	Z	Z	B
4				C	B	Z
5					C	B
6						C

	C	B	Z	B	Z	Z
1						
2		C	B	Z	B	Z
3			C	Z	Z	B
4				C	B	Z
5					C	B
6						C
	1	2	3	4	5	6

(1,2)	□	(1,1)	C
(2,2)	C	(1,2)	□
(1,2)	□	(3,2)	□
(4,2)	□	(1,4)	□
(5,2)	□	(1,5)	□
(6,2)	□	(1,6)	□

	C	B	Z	B	Z	Z
1						
2		C	B	Z	B	Z
3			C	Z	Z	B
4				C	B	Z
5					C	B
6						C
1	2	3	4	5	6	

(1,2)	□	(1,1)	C
(2,2)	C	(1,2)	□
(1,2)	□	(3,2)	□
(4,2)	□	(1,3)	C
(5,2)	□	(1,4)	□
(6,2)	□	(1,5)	C
		(1,6)	C
(1,2)	□	(3,1)	C
(2,2)	C	(3,2)	□
(3,2)	□	(3,2)	C
(4,2)	□	(3,3)	C
(5,2)	□	(3,4)	C
(6,2)	□	(3,5)	C
		(3,6)	□

1	C	B	Z	B	Z	Z
2		C	B	Z	B	Z
3	B		C	Z	Z	B
4		B		C	B	Z
5	B		B		C	B
6		B				C

Diagram illustrating a mapping between a 6x6 input grid and a 6x6 output grid.

The input grid (left) has rows labeled 1 to 6 and columns labeled 1 to 6. The output grid (right) has rows labeled 1 to 6 and columns labeled 1 to 6. Cells are colored according to their values:

- (C, B, Z) are cyan
- (B, Z) are blue
- (C) is white
- (R, M, G, N) are gray/black
- (V) is green
- (C, R, M) are orange
- (B, R) are pink
- (N, G) are purple

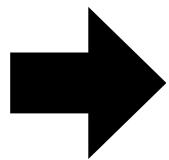
Arrows indicate specific mappings:

- An arrow points from cell (1,2) in the input grid to cell (1,1) in the output grid.
- An arrow points from cell (3,2) in the input grid to cell (3,3) in the output grid.

Below the grids, two sets of coordinate pairs are listed:

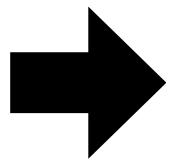
- Top set: (1,2), (1,1), (2,2), (1,2), (3,2), (1,3), (4,2), (1,4), (5,2), (1,5), (6,2), (1,6)
- Bottom set: (1,2), (3,1), (2,2), (3,2), (3,2), (3,3), (4,2), (3,4), (5,2), (3,5), (6,2), (3,6)

1	C	B	Z	B	Z	Z
2		C	B	Z	B	Z
3			C	Z	Z	B
4			C	B	Z	
5				C	B	
6					C	
	1	2	3	4	5	6
	7	8	9	10	11	12



1	C	R	M	G	N	M
2		B	R	N	V	N
3			C	M	N	G
4				C	R	M
5					B	R
6					C	
	1	2	3	4	5	6
	7	8	9	10	11	12

7	C	B	B	Z	Z	Z
8		C	B	Z	Z	Z
9			C	B	Z	Z
10				C	B	B
11				C	B	
12					C	
	1	2	3	4	5	6
	7	8	9	10	11	12



7	C	G	R	N	M	M
8		C	R	N	M	M
9			B	V	N	N
10				B	R	R
11					C	G
12					C	
	1	2	3	4	5	6
	7	8	9	10	11	12

1	C	R	M	G	N	M
2	B	R	N	V	N	N
3		C	M	N		G
4			C	R	M	M
5				B	R	R
6					C	

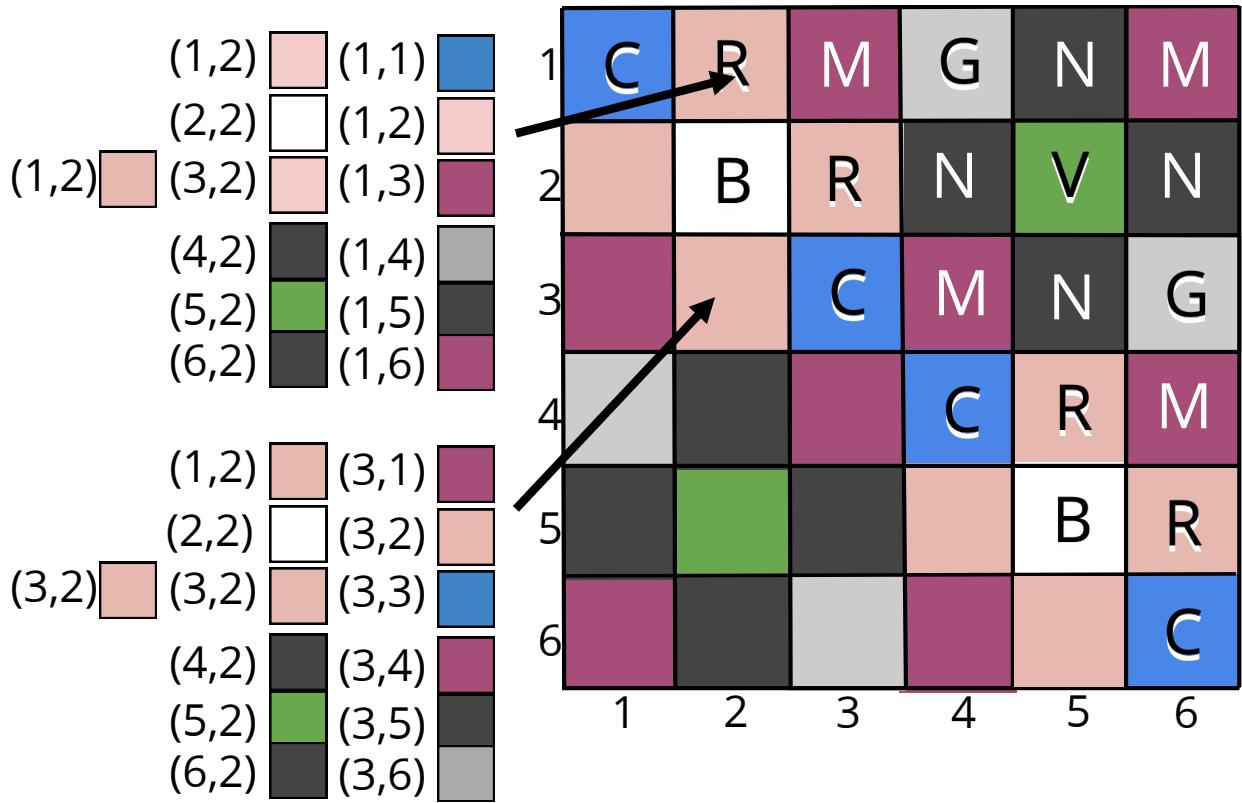
1	C	R	M	G	N	M
2	B	R	N	V	N	
3		C	M	N		G
4			C	R	M	
5				B	R	
6					C	

(1,2)	(1,1)	(1,2)
(2,2)	(1,2)	(1,2)
(1,2)	(3,2)	(1,3)
(4,2)	(1,4)	(1,4)
(5,2)	(1,5)	(1,5)
(6,2)	(1,6)	(1,6)

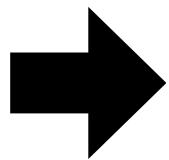
1	C	R	M	G	N	M
2	B	R	N	V	N	
3		C	M	N		G
4			C	R	M	
5				B	R	
6					C	
	1	2	3	4	5	6

(1,2)	(1,1)	(1,2)
(2,2)	(1,2)	(1,2)
(1,2)	(3,2)	(1,3)
(4,2)	(1,4)	(1,4)
(5,2)	(1,5)	(1,5)
(6,2)	(1,6)	(1,6)
(1,2)	(3,1)	(3,1)
(2,2)	(3,2)	(3,2)
(3,2)	(3,2)	(3,3)
(4,2)	(3,4)	(3,4)
(5,2)	(3,5)	(3,5)
(6,2)	(3,6)	(3,6)

1	C	R	M	G	N	M
2		B	R	N	V	N
3			C	M	N	G
4				C	R	M
5					B	R
6						C
	1	2	3	4	5	6

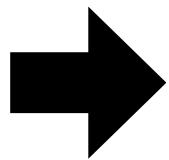


1	C	R	M	G	N	M
2		B	R	N	V	N
3			C	M	N	G
4			C	R	M	
5				B	R	
6					C	



1	C	R	M	G	N	M
2		B	R	N	V	N
3			C	M	N	G
4			C	R	M	
5				B	R	
6					C	

7	C	G	R	N	M	M
8		C	R	N	M	M
9			B	V	N	N
10			B	R	R	
11			C	G		
12					C	



7	C	G	R	N	M	M
8		C	R	N	M	M
9			B	V	N	N
10			B	R	R	
11			C	G		
12					C	

1	C	R	M	G	N	M
2		B	R	N	V	N
3			C	M	N	G
4			C	R	M	
5				B	R	
6						C
	1	2	3	4	5	6
	7	8	9	10	11	12

7	C	G	R	N	M	M
8		C	R	N	M	M
9			B	V	N	N
10			B	R	R	
11				C	G	
12						C

$$C(G) = \{$$

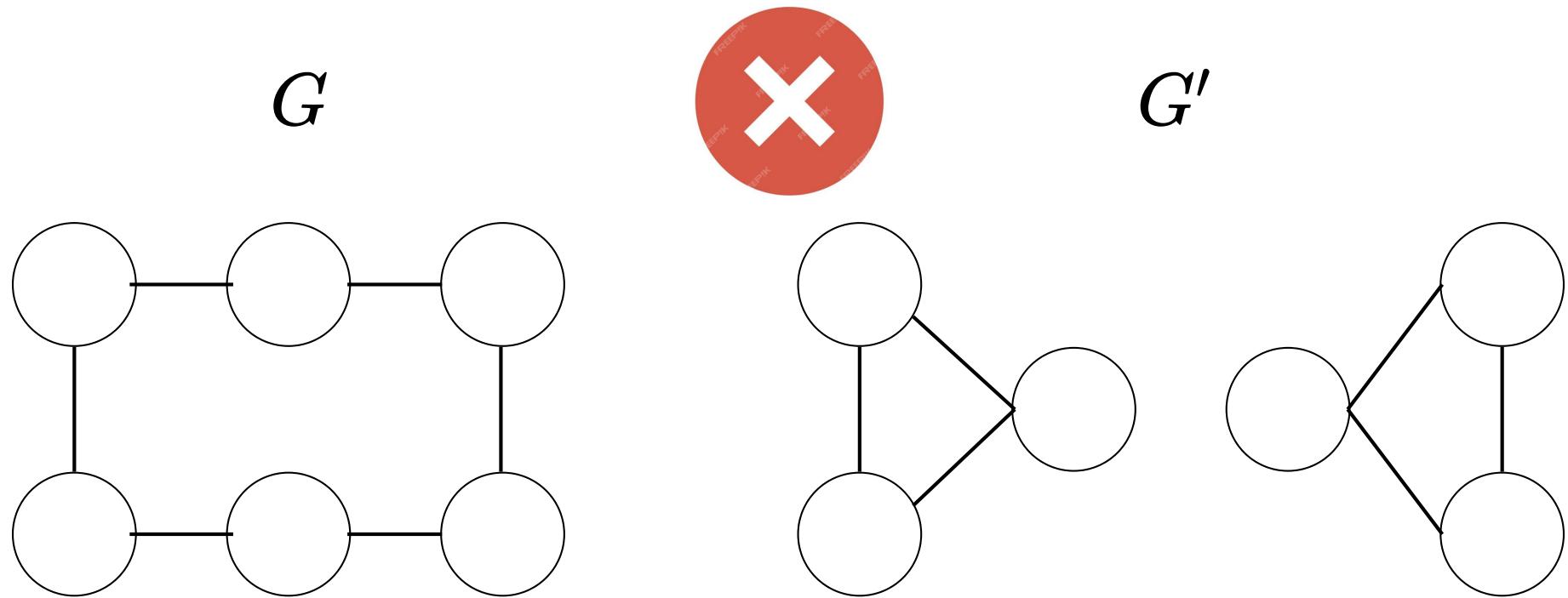
$$\begin{array}{l} C \times 4, \quad R \times 8, \quad M \times 8, \\ G \times 4, \quad N \times 8, \quad B \times 2, \\ V \times 2 \end{array} \}$$



$$C(G') = \{$$

$$\begin{array}{l} C \times 4, \quad R \times 8, \quad M \times 8, \\ G \times 4, \quad N \times 8, \quad B \times 2, \\ V \times 2 \end{array} \}$$

# Test de isomorfismo 2-WL



¿Qué resultado da 2-WL sobre estos grafos?

**1-WL es equivalente a 2-WL**

(Weisfeiller & Lehman, 1969)

# Algoritmo $k$ -WL

**Input:**  $G = (V, E)$

0.  $C \leftarrow \text{ColoreoInicial}(G)$

2. **repeat**

3.    $C^{old} \leftarrow C$

4.   **for each**  $\bar{v} \in V^k$  **do**

5.     **for**  $i = 1$  to  $k$  **do**

6.        $\chi[i] \leftarrow \{[C^{old}[\bar{w}] \mid \bar{w} \in N_i(\bar{v})]\}$

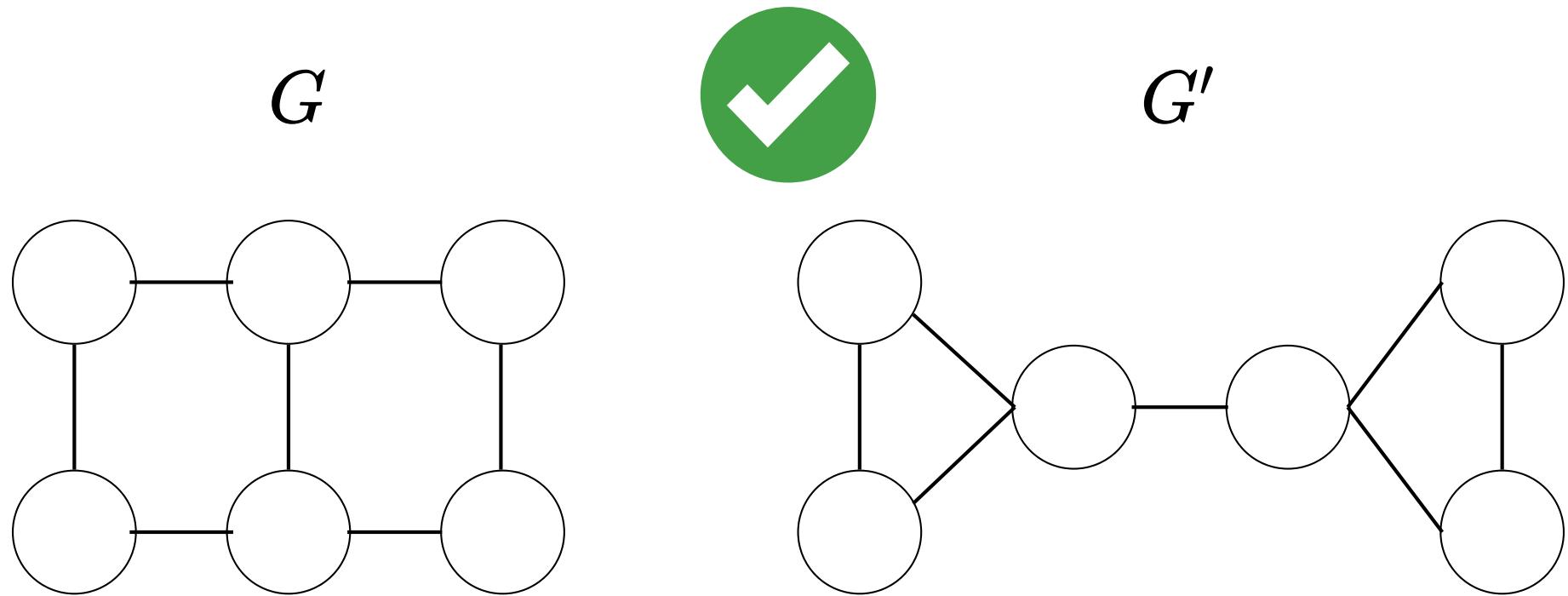
7.      $C[\bar{v}] \leftarrow \text{hash}\left(c^{old}[\bar{v}], \chi[1], \dots, \chi[k]\right)$

8. **until not**  $\sim_C$  es un refinamiento estricto de  $\sim_{C^{old}}$

9. **return**  $C$

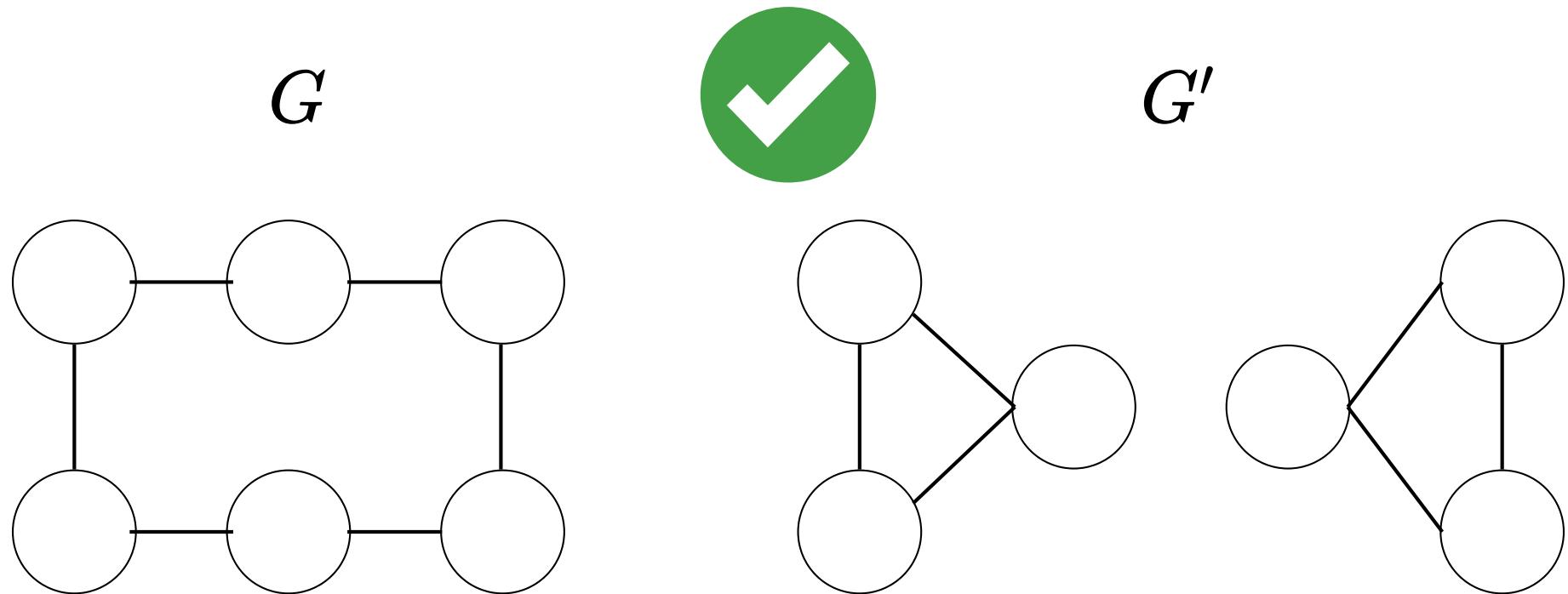
$$N_i(\bar{v}) = \{(v_1, \dots, v_{i-1}, m, v_{i+1}, \dots, v_k) \\ m \in V\}$$

# Test de isomorfismo 3-WL



¿Qué resultado da 3-WL sobre estos grafos?

# Test de isomorfismo 3-WL



¿3-WL también da el resultado correcto en este caso?

# Algoritmo $k$ -folklore-WL ( $k$ -WL especial)

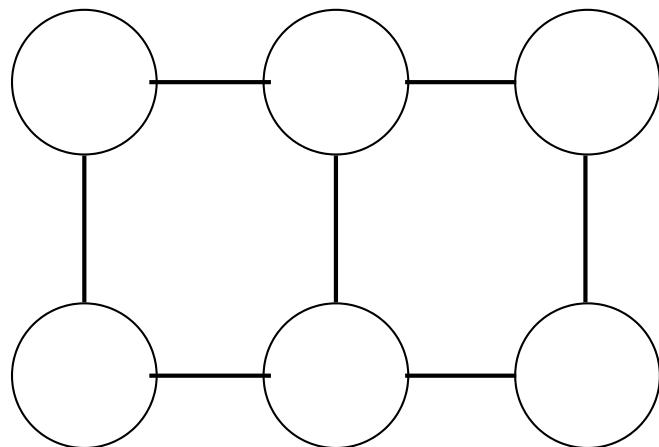
# Algoritmo $k$ -folklore-WL

**Input:**  $G = (V, E)$

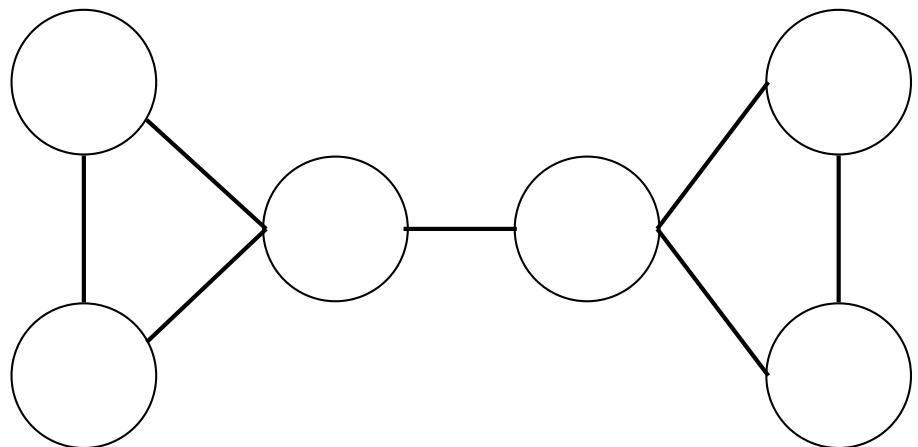
1.  $C \leftarrow \text{ColoreoInicial}(G)$
2. **repeat**
3.    $C^{old} \leftarrow C$
4.   **for each**  $(v_1, \dots, v_k) \in V^k$  **do**
5.     **for each**  $w \in V$  **do**
6.       **for**  $j = 1$  **to**  $k$  **do**
7.          $\Delta_w \leftarrow [C^{old}(w, v_2, \dots, v_k), \dots, C^{old}(v_1, \dots, v_{k-1}, w)]$
8.          $C[\bar{v}] \leftarrow \text{hash}\left(C^{old}[\bar{v}], \{\Delta_w \mid w \in V\}\right)$
9.   **until not**  $\sim_C$  es un refinamiento estricto de  $\sim_{C^{old}}$
10. **return**  $C$

# Test de isomorfismo 2-folklore-WL

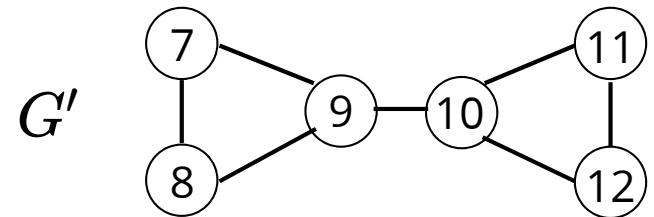
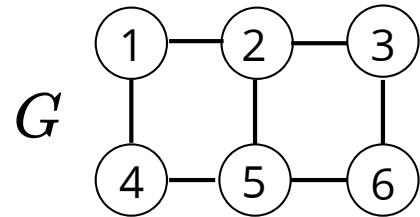
$G$



$G'$



## ColoreoInicial( $G$ )

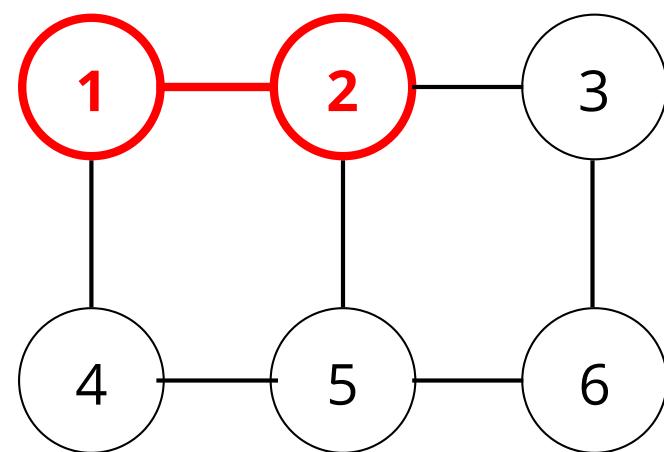


	1	2	3	4	5	6
1	C	B	Z	B	Z	Z
2		C	B	Z	B	Z
3	B		C	Z	Z	B
4		B		C	B	Z
5	B			C	B	
6	B		B			C

	7	8	9	10	11	12
7	C	B	B	Z	Z	Z
8		C	B	Z	Z	Z
9			C	B	Z	Z
10	B			C	B	B
11	B			C	B	
12	B					C

1	C	B	Z	B	Z	Z
2		C	B	Z	B	Z
3			C	Z	Z	B
4				C	B	Z
5					C	B
6						C

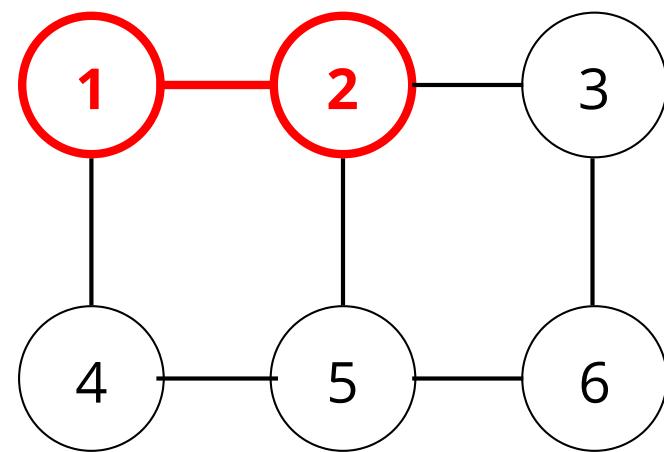
(2,1)



	C	B	Z	B	Z	Z
1						
2		C	B	Z	B	Z
3			C	Z	Z	B
4				C	B	Z
5					C	B
6						C
1	2	3	4	5	6	

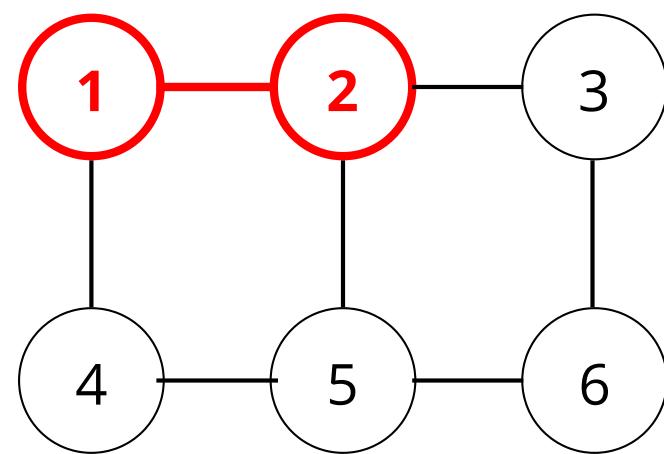
(2,1)

(1,1)   
 (2,1)



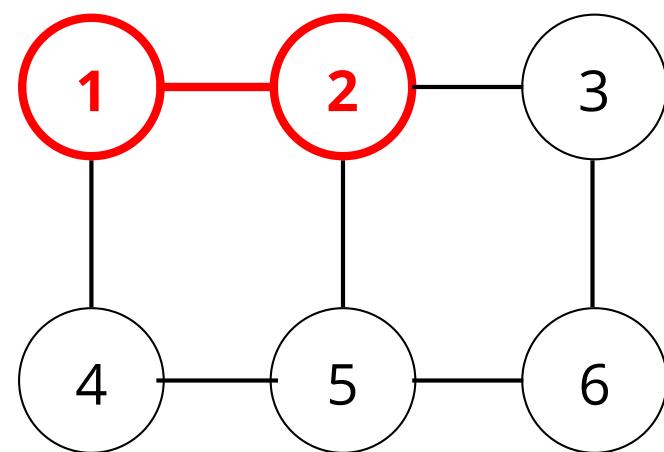
	C	B	Z	B	Z	Z
1						
2		C	B	Z	B	Z
3			C	Z	Z	B
4				C	B	Z
5					C	B
6						C
1	2	3	4	5	6	

(2,1)   
 (1,1)  (2,1)   
 (2,1)  (2,2)



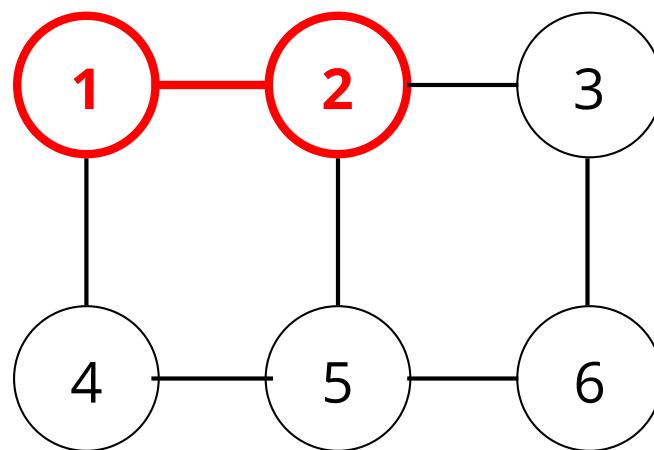
	C	B	Z	B	Z	Z
1						
2		C	B	Z	B	Z
3			C	Z	Z	B
4				C	B	Z
5					C	B
6						C
1	2	3	4	5	6	

$(2,1)$    
 $(1,1)$   $(2,1)$   $(3,1)$    
 $(2,1)$   $(2,2)$   $(2,3)$



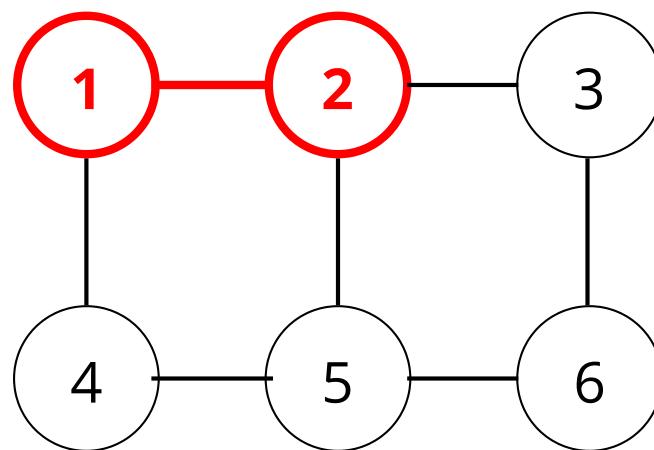
1	C	B	Z	B	Z	Z
2		C	B	Z	B	Z
3			C	Z	Z	B
4				C	B	Z
5					C	B
6						C
	1	2	3	4	5	6

$(2,1)$    
 $(1,1)$   $(2,1)$   $(3,1)$    
 $(2,1)$   $(2,2)$   $(2,3)$    
 $(4,1)$    
 $(2,4)$

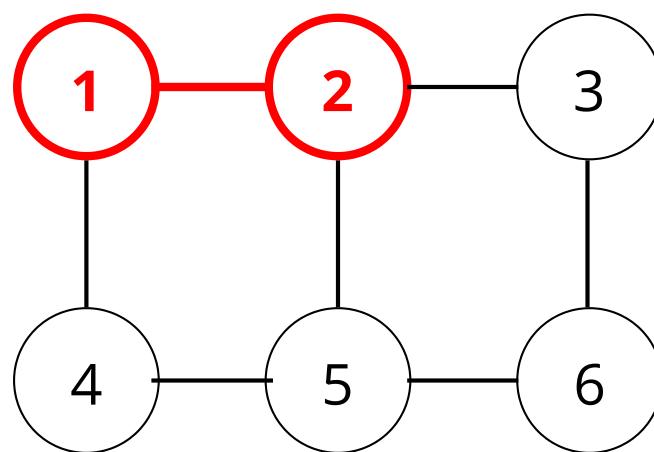
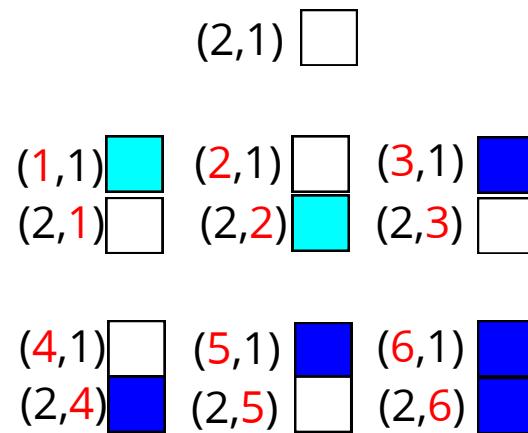


1	C	B	Z	B	Z	Z
2		C	B	Z	B	Z
3			C	Z	Z	B
4				C	B	Z
5					C	B
6						C
	1	2	3	4	5	6

$(2,1)$    
 $(1,1)$   $(2,1)$   $(3,1)$    
 $(2,1)$   $(2,2)$   $(2,3)$    
 $(4,1)$   $(5,1)$    
 $(2,4)$   $(2,5)$



1	C	B	Z	B	Z	Z
2		C	B	Z	B	Z
3			C	Z	Z	B
4				C	B	Z
5					C	B
6						C
	1	2	3	4	5	6

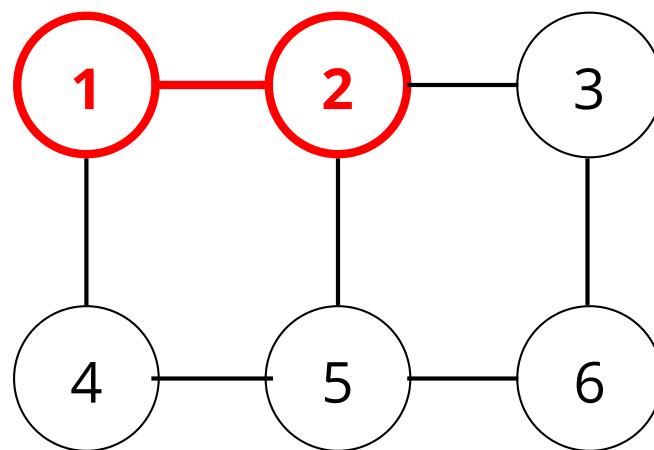


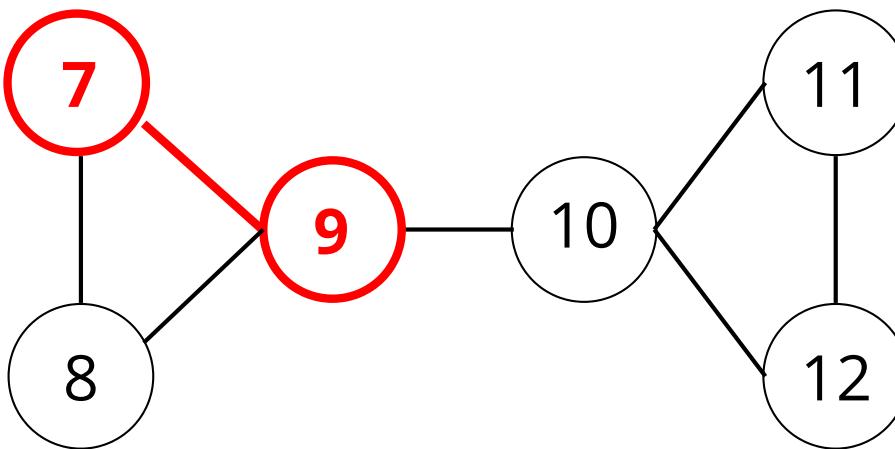
	C	B	Z	B	Z	Z
1						
2		C	B	Z	B	Z
3			C	Z	Z	B
4				C	B	Z
5					C	B
6						C
	1	2	3	4	5	6

(2,1)   
 (1,1)  (2,1)  (3,1)   
 (2,1)  (2,2)  (2,3)   
 (4,1)  (5,1)  (6,1)   
 (2,4)  (2,5)  (2,6)



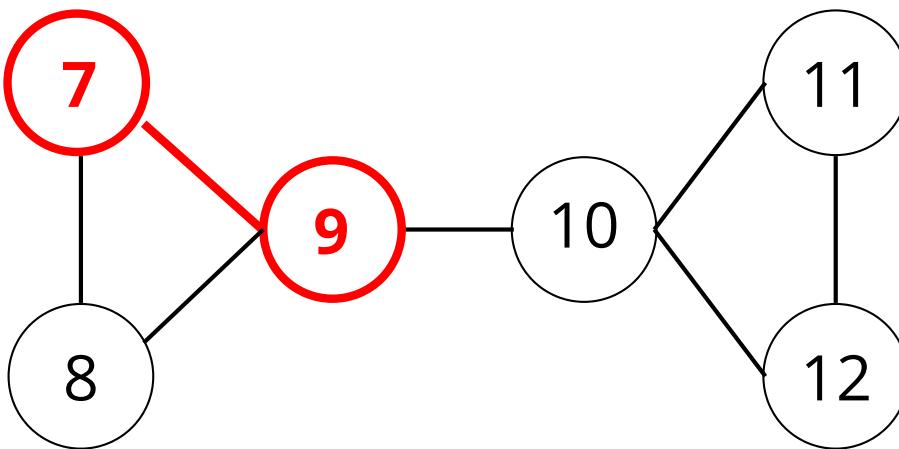
	C	R	J	G	N	M
1						
2		B	R	N	V	N
3				C	M	N
4					C	R
5					B	R
6					C	
	1	2	3	4	5	6





	7	8	9	10	11	12
7	C	B	B	Z	Z	Z
8		C	B	Z	Z	Z
9			C	B	Z	Z
10				C	B	B
11					C	B
12						C

$(9,7)$    
 $(7,7)$   $(8,7)$   $(9,7)$    
 $(9,7)$   $(9,8)$   $(9,9)$    
 $(10,7)$   $(11,7)$   $(12,7)$    
 $(9,10)$   $(9,11)$   $(9,12)$



	7	8	9	10	11	12
7	C	B	B	Z	Z	Z
8	C	B	Z	Z	Z	Z
9		C	B	Z	Z	Z
10			C	B	B	
11				C	B	
12					C	

$(9,7)$    
 $(7,7)$    
 $(9,7)$    
 $(10,7)$    
 $(9,10)$    
 $(8,7)$    
 $(9,8)$    
 $(9,9)$    
 $(11,7)$    
 $(9,11)$    
 $(12,7)$    
 $(9,12)$

	7	8	9	10	11	12
7	C	E	A	O	M	M
8	C	A	O	M	M	M
9			B	V	O	O
10				B	A	A
11				C	E	
12					C	

C	B	Z	B	Z	Z
	C	B	Z	B	Z
		C	Z	Z	B
			C	B	Z
				C	B
					C

C	B	B	Z	Z	Z
	C	B	Z	Z	Z
		C	B	Z	Z
			C	B	B
				C	B
					C

Diagram illustrating a mapping from a 6x6 grid to a 12x12 grid.

The 6x6 grid (left) has rows labeled 1 to 6 and columns labeled 1 to 12. The 12x12 grid (right) has rows labeled 1 to 12 and columns labeled 7 to 12. Colored squares indicate specific mappings:

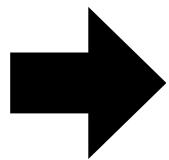
- (2,1) → (2,1)
- (1,1) → (1,1)
- (2,1) → (2,1)
- (2,2) → (2,1)
- (2,3) → (3,1)
- (4,1) → (4,1)
- (2,4) → (2,4)
- (2,5) → (2,5)
- (2,6) → (2,6)
- (5,1) → (5,1)
- (6,1) → (6,1)

Diagram illustrating a mapping from a 12x12 grid to a 12x12 grid.

The 12x12 grid (left) has rows labeled 7 to 12 and columns labeled 7 to 12. The 12x12 grid (right) has rows labeled 7 to 12 and columns labeled 7 to 12. Colored squares indicate specific mappings:

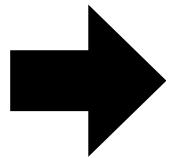
- (9,7) → (9,7)
- (7,7) → (7,7)
- (9,7) → (9,7)
- (8,7) → (8,7)
- (9,8) → (9,8)
- (9,9) → (9,9)
- (10,7) → (10,7)
- (9,10) → (9,10)
- (11,7) → (11,7)
- (9,11) → (9,11)
- (12,7) → (12,7)
- (9,12) → (9,12)

1	C	B	Z	B	Z	Z
2		C	B	Z	B	Z
3			C	Z	Z	B
4				C	B	Z
5					C	B
6						C
	1	2	3	4	5	6
	7	8	9	10	11	12



1	C	R	J	G	N	M
2		B	R	N	V	N
3			C	M	N	G
4				C	R	J
5					B	R
6						C
	1	2	3	4	5	6
	7	8	9	10	11	12

7	C	B	B	Z	Z	Z
8		C	B	Z	Z	Z
9			C	B	Z	Z
10				C	B	B
11					C	B
12						C
	1	2	3	4	5	6
	7	8	9	10	11	12



7	C	E	A	O	M	M
8		C	A	O	M	M
9			B	V	O	O
10				B	A	A
11					C	E
12						C
	1	2	3	4	5	6
	7	8	9	10	11	12

	C	R	J	G	N	M
1						
2	B	R	N	V	N	
3		C	M	N	G	
4		C	R	J		
5		B	R			
6				C		
7	1	2	3	4	5	6
8	7	8	9	10	11	12

	C	E	A	O	M	M
7						
8	C	A	O	M	M	
9		B	V	O	O	
10		B	A	A		
11		C	E			
12				C		
13						
14						

Diagram illustrating a 6x6 matrix transformation from the initial state to a final state.

Initial State (Left):

	C	R	J	G	N	M
1						
2	B	R	N	V	N	
3		C	M	N	G	
4		C	R	J		
5		B	R			
6				C		
7	1	2	3	4	5	6
8	7	8	9	10	11	12

Final State (Right):

	C	R	J	G	N	M
1						
2	B	R	N	V	N	
3		C	M	N	G	
4		C	R	J		
5		B	R			
6				C		
7	1	2	3	4	5	6
8	7	8	9	10	11	12

Color Legend:

- (1,1) Blue
- (2,1) Brown
- (3,1) Orange
- (4,1) Grey
- (5,1) Dark Grey
- (6,1) Purple
- (7,1) Green
- (8,1) Yellow
- (9,1) Red
- (10,1) Light Blue
- (11,1) Magenta
- (12,1) Cyan

Coordinate Pairs:

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

Diagram illustrating a 6x6 matrix transformation from the initial state to a final state.

Initial State (Left):

	T	E	A	O	Z	Z
7						
8	T	A	O	Z	Z	
9		S	F	O	O	
10		S	A	A		
11		T	E			
12				T		

Final State (Right):

	T	E	A	O	Z	Z
7						
8	T	A	O	Z	Z	
9	S	F	O	O	O	
10	S	A	A			
11	T	E				
12				T		

Color Legend:

- (1,1) Cyan
- (2,1) Green
- (3,1) Yellow
- (4,1) Red
- (5,1) Blue
- (6,1) Dark Blue
- (7,1) Magenta
- (8,1) Light Blue
- (9,1) Orange
- (10,1) Light Green
- (11,1) Purple
- (12,1) Cyan

Coordinate Pairs:

- (9,7), (8,7)
- (9,7), (9,8)
- (9,8), (9,9)
- (10,7), (11,7)
- (9,10), (9,11)
- (9,11), (9,12)
- (12,7), (12,8)

1	C	R	J	G	N	M
2	B	R	N	V	N	
3	C	M	N	G		
4	C	R	J			
5	B	R				
6	C					

1    2    3    4    5    6  
7    8    9    10    11    12

7	T	E	A	O	Z	Z
8	T	A	O	Z	Z	
9	S	F	O	O		
10	S	A	A			
11	T	E				
12	T					

$$C(G) = \{$$

<span style="background-color: #0070C0; color: white; padding: 5px;">C</span>	x 4,	<span style="background-color: #C0A0A0; color: white; padding: 5px;">R</span>	x 8,	<span style="background-color: #FFA000; color: white; padding: 5px;">J</span>	x 4,
<span style="background-color: #A0A0C0; color: black; padding: 5px;">G</span>	x 4,	<span style="background-color: #606060; color: white; padding: 5px;">N</span>	x 8,	<span style="background-color: #804080; color: white; padding: 5px;">M</span>	x 4,
<span style="border: 1px solid black; padding: 5px;">B</span>	x 2,	<span style="background-color: #609040; color: white; padding: 5px;">V</span>	x 2		}



$$C(G') = \{$$

<span style="background-color: #00FFFF; color: black; padding: 5px;">T</span>	x 4,	<span style="background-color: #00FF00; color: black; padding: 5px;">E</span>	x 4,	<span style="background-color: #FFDAB9; color: black; padding: 5px;">A</span>	x 8,
<span style="background-color: #FF0000; color: black; padding: 5px;">O</span>	x 8,	<span style="background-color: #0000FF; color: black; padding: 5px;">Z</span>	x 8,	<span style="background-color: #4682B4; color: black; padding: 5px;">S</span>	x 2,
<span style="background-color: #FF00FF; color: black; padding: 5px;">F</span>	x 2				}

**$(k - 1)$ -folklore-WL es equivalente  
a  $k$ -WL**

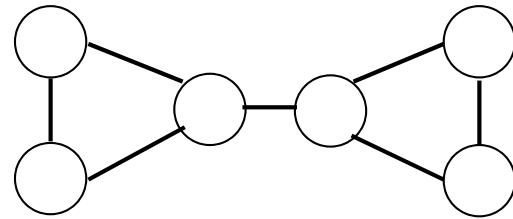
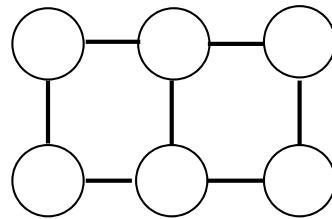
(Cai, Furer & Immerman, 1992)

# ¿Dónde estamos?

- El test WL debe aplicarse a la unión disjunta de los dos grafos de entrada.
- Al incrementar  $k$ , el algoritmo  $k$ -WL tiene un mayor poder de distinguir dos grafos no-isomorfos.
- 1-WL es equivalente a 2-WL.
- $(k - 1)$ -folklore-WL es equivalente a  $k$ -WL.

# ¿Cuál es el $k$ necesario?

- No hay un valor fijo  $k$  tal que  $k$ -WL decide correctamente si dos grafos son isomorfos.
- 4-WL decide correctamente si dos grafos planos son isomorfos (Kiefer, Ponomarenko, Schweitzer, 2019)



¿Se puede extender este resultado a otras clases de grafos?