

Traza de Depuración: Floyd-Warshall con Ciclo Negativo

El Bug

El código original ejecutaba las relajaciones correctamente pero **no verificaba la diagonal principal** al final. En Floyd-Warshall, $\text{dist}[i][i]$ representa la distancia mínima para ir de i a i . Inicialmente es 0. Si al final del algoritmo $\text{dist}[i][i] < 0$, significa que existe un camino que sale de i y vuelve a i con costo negativo (un ciclo negativo).

Traza Paso a Paso (Input: 0->1:2, 1->2:-1, 2->0:-4)

Inicialización (k=-1):

	0	1	2
0	[0, 2, inf]		
1	[inf, 0, -1]		
2	[-4, inf, 0]		

Iteración k=0 (Pivote nodo 0):

Mejoramos caminos pasando por 0.

$$\text{dist}[2][1] = \min(\text{inf}, \text{dist}[2][0] + \text{dist}[0][1]) = -4 + 2 = -2$$

	0	1	2
0	[0, 2, inf]		
1	[inf, 0, -1]		
2	[-4, -2, 0]		

Iteración k=1 (Pivote nodo 1):

Mejoramos caminos pasando por 1.

$$\text{dist}[0][2] = \min(\text{inf}, \text{dist}[0][1] + \text{dist}[1][2]) = 2 + (-1) = 1$$

$\text{dist}[2][2] = \min(0, \text{dist}[2][1] + \text{dist}[1][2]) = -2 + (-1) = -3$ -- ¡Diagonal negativa!

	0	1	2
0	[0, 2, 1]		
1	[inf, 0, -1]		
2	[-4, -2, -3]		

Iteración k=2 (Pivote nodo 2):

Mejoramos caminos pasando por 2.

$\text{dist}[0][0] = \min(0, \text{dist}[0][2] + \text{dist}[2][0]) = 1 + (-4) = -3$ <-- ¡Diagonal negativa!

$\text{dist}[0][1] = \min(2, \text{dist}[0][2] + \text{dist}[2][1]) = 1 + (-2) = -1$

$\text{dist}[1][0] = \min(\infty, \text{dist}[1][2] + \text{dist}[2][0]) = -1 + (-4) = -5$

$\text{dist}[1][1] = \min(0, \text{dist}[1][2] + \text{dist}[2][1]) = -1 + (-2) = -3$ <-- ¡Diagonal negativa!

	0	1	2
0	-3, -1,	1	
1	-5, -3,	-1	
2	-4, -2,	-3	

Verificación Final (Corrección)

El bucle de verificación revisa $\text{dist}[i][i]$:

- $\text{dist}[0][0] = -3 (< 0) \rightarrow \text{ValueError: Ciclo negativo detectado}$