

Video38 (1)

Conceptos Principales

1. a
2. b
3. c

Notas

En este video se explica cómo construir **la matriz de caminos** de un grafo. La idea es simple: la matriz de adyacencia solo te dice si hay conexión directa entre dos nodos; la matriz de caminos te dice si **existe algún camino**, directo o pasando por otros nodos.

Por ejemplo: si no hay flecha directa de A a D, la matriz de adyacencia pone un 0. Pero si existe un camino $A \rightarrow B \rightarrow D$ aunque sea por pasos intermedios, la matriz de caminos debería poner un 1.

La pregunta que queremos resolver es: **¿se puede llegar del vértice i al vértice j aunque sea con varios brincos?**

Cómo se construye la matriz de caminos (cierre transitivo)

El método visto en este video usa operaciones tipo “álgebra booleana” aplicadas a matrices.

Para entenderlo, hay que recordar:

- En matrices booleanas, la **multiplicación** se interpreta como un **AND**.
- La **suma** se interpreta como un **OR**.

Lo que se hace es lo siguiente:

1. **Se copia la matriz de adyacencia en una nueva matriz**, que será la matriz de caminos inicial.
2. Luego, se “simulan” caminos cada vez más largos haciendo **multiplicaciones de matrices**:
 - A^1 representa caminos de 1 brinco (la adyacencia normal).
 - A^2 representa caminos de 2 brincos.
 - A^3 representa caminos de 3 brincos.
 - ... y así hasta llegar a n multiplicaciones (n = cantidad de nodos).
3. Si en alguna de esas matrices aparece un 1 en la posición (i, j) , significa que existe un camino desde i hasta j con esa cantidad de brincos.
4. La matriz final de caminos marca 1 si **hay al menos un camino** de cualquier longitud razonable ($\leq n$).

La lógica detrás es:

para saber si i llega a j con 2 pasos, buscás si existe algún k tal que i llegue a k **y** k llegue a j.

Ese “y” es el AND; el “algún k” termina siendo un OR.

Todo eso lo hacen las multiplicaciones de matrices booleanas.

Por qué sirve

Este método es útil cuando querés responder preguntas como:

- ¿Existe alguna ruta de un nodo a otro?
- ¿Se puede alcanzar un vértice sin importar cuántos pasos sean?
- ¿Cuántos pasos mínimos se necesitarían? (si además contás en qué multiplicación apareció el 1)

En programación en C, este algoritmo se puede implementar usando las mismas rutinas básicas de multiplicación de matrices que se ven en cursos introductorios, solo que usando operaciones booleanas en vez de sumas y productos normales. Es más práctico usar matrices (no listas de adyacencia) porque el algoritmo depende directamente de combinaciones fila–columna.