

Laboratorio Nro. 5: Implementación de Grafos

Agustín Rico
Universidad Eafit
Medellín, Colombia
aricop@eafit.edu.co

Santiago Cano
Universidad Eafit
Medellín, Colombia
scanof@eafit.edu.co

3) Simulacro de preguntas de sustentación de Proyectos

1. ...
2. Para la matriz de adyacencia, se establecen como columnas los puntos de partida y como filas los puntos de llegada. Luego, la matriz se llenará bajo el criterio de si existe o no una ruta que vaya desde el nodo correspondiente a la columna, hacia el nodo de la fila. En ese caso se pondrá un 1, o si se quiere, la magnitud o distancia de dicho camino. En caso de que tal camino no exista en el grafo, se pondrá un cero. El grafo con listas de adyacencia se crea bajo un conjunto de listas. Cada primer elemento de cada lista representa un nodo, y todos los elementos que pertenezcan es porque existe una ruta desde el nodo primero hasta el nodo en cuestión.
3. La implementación con matrices es más favorable para grafos los cuales sus nodos están conectados casi que todos, o muchos, entre sí. Es decir, no tiene sentido realizar una implementación con matrices si la mayoría de sus celdas van a ser ceros. Las listas enlazadas son justamente ideales para dicho caso. Si un grafo tiene muchos nodos pero no tantas conexiones entre ellos, entonces es conveniente usar listas, puesto que sólo se utilizará memoria en caso de que una ruta exista, y no se va a ocupar con ceros, como sería el caso de una matriz.
4. Listas de adyacencia. Es claro que la mayoría de calles no se conectan las unas con las otras, incluso, únicamente se conectan con las adyacentes. Dado este caso, es mejor emplear listas, en cuyas su primer elemento sea una calle cualquiera y sus elementos siguientes serían las calles con las que la calle en cuestión se conecta. Así, a pesar de que para almacenar cada lista correspondiente a cada calle se tendría que tener un arreglo tan largo como calles hay en Medellín, es mucho mejor que tener ese arreglo al cuadrado, es decir, una matriz cuadrada de la misma longitud del arreglo, que estaría, en su mayoría, llena de ceros.
5. En un caso como el de una red social es muchísimo mejor una lista enlazada. Pues no tiene sentido crear una matriz cuadrada de longitud igual al número de usuarios de Facebook, si en cada fila que representa a una persona, se encuentran muy pocos espacios ocupados con un 1, en comparación con los espacios llenos de ceros. En resumen, utilizar una matriz sería un gasto de memoria, no sólo gigante, sino también muy desperdiciada.
6. En este caso creemos que es mejor la implementación con matrices. En un hogar, por ejemplo, no hay un número muy grande de dispositivos que accedan a internet. De esta manera, no es problema ni supone un desuso de memoria la implementación con matrices. Incluso, puesto que el router tendrá que calcular la ruta más corta en el menor tiempo posible, sabemos que en una matriz no tan grande, la eficiencia de búsqueda es mucho mayor en una matriz que en una lista enlazada.
7. ...

8. ...

4) Simulacro de Parcial

	0	1	2	3	4	5	6	7
0				1	1			
1	1		1			1		
2					1		1	
3								1
4			1					
5								
6			1					
7								

- 1.
2. $0 \rightarrow [3,4]; \quad 1 \rightarrow [0, 2, 5]; \quad 2 \rightarrow [4, 6]; \quad 3 \rightarrow [7]; \quad 4 \rightarrow [2]; \quad 5 \rightarrow []; \quad 6 \rightarrow [2];$
3. b)