11/07/19

Grafos:

Grafos no dirigidos: son grafos en los que las aristas son todas bidireccionales

Matriz de adyacencia:

Matriz de nxn, con n a cantidad de nodos del grafo. En la posicion (i,j) tiene un 1(true ) si hay una arista entre los nodos i y j, o 0 (false) caso contrario.

Es O(1) decir si dos nodos estan conectados o no.

Si queres ver los vecinos (quien esta conectado con quien), es lineal (poco optimo en tiempo y memoria)

Lista de adyacencia:

Vector de vectores enteros. En el vector i-esimo tiene el numero j sii hay una arista entre los nodos j e i.

Se usa para BFS Y DFS.

Grafos esparsos: grafos que tienen pocas aristas. La lista de adyacencia es mas eficiente para este tipo de grafos.

(hay codigo util en las diapositivas)

Distancia:

Un camino es un recorrido de aristas donde el final de una encaja con el principio de la otra

(menor distancia es menor cantidad de aristas recorridas)

Encontrar la distancia entre dos nodos de un grafo: BFS: (fifo)

(Busqueda en anchura)

Cada nodo es procesado una sola vez. Cada arista es vista una sola vez (o una vez por sentido para no-dirigidos)

La complejidad es entonces: O(|E|)

DFS: (filo)

Es un BFS que en vez de usar una queue usa un stack

Orden topologico (como un flujo de agua)

Caso particular de grafos dirigidos sin ciclos. El agua “cae” del de mas arriba al de mas abajo. Puede haber nodos en paralelo. Se usa para modelizar cosas que el orden esta necesariamente concatenado (no se puede paralelizar)

Si un grafo no tiene ciclos, podemos encontrar el orden topologico asi;

Llamamos a DFS para todos los nodos (excepto los que visitamos en un paso anterior)

Cuando termina el DFS desde cada nodo, agregamos el nodo al principio del orden topologico.

Grafos ponderados:

Los pesos pueden ser naturales, enteros o reales

Conviene poner el peso en la primera componente del par y el nodo en la segunda.

Dijkstra:

Dijkstra no funciona con pesos negativos

Complejidad: O(|v|²)

Usando una cola de prioridad para elegir el nodo de menor distancia la complejidad pasa a ser O(|E| log(|E|))

Recordar if() continue;

Para grafos con pesos negativos, convieneusar el algoritmo de Bellman-Ford o Floyd-Warshall (es cubico si los nodos son del orden de mil)

pastebin.com/LkWjyixi

Arbol: Grafo conexo sin ciclos

|E| = |V|-1 (cant aristas = cant nodos -1)

Union find: estructura que sirve para “pegar” objetos (es muy rapida ~O(1))

Soporta estas operaciones:

find(x)

union(x,y)

Arbol generador minimo

Algoritmo de Kruskal (es casi como meter greedy) (O (n logn))