

Práctico 2 ALGABO: Grafos y DFS.

Mauricio Velasco

1. Proponga un algoritmo para listar las componentes conexas de un grafo no dirigido. Demuestre matemáticamente que el algoritmo que propone produce la respuesta correcta.
2. Dibuje un grafo dirigido G cuya versión no-dirigida sea conexo y que tenga tres componentes fuertemente conexas distintas (Dato curioso: Las componentes fuertemente conexas de un grafo dirigido se pueden calcular en tiempo lineal usando DFS dos veces mediante el *algoritmo de Kosaraju*, estudiar este algoritmo podría ser un proyecto de interés).
3. Demuestre las siguientes afirmaciones:
 - a) En todo grafo dirigido G tenemos $\sum_{v \in V(G)} n_v = |E|$ donde $n_v := |\text{Out}(v)|$. Cómo cambia este enunciado para grafos no dirigidos?
 - b) Todo par de vértices en un árbol está unido por un único camino de longitud mínima.
4. Visitamos todos los vértices del grafo G no dirigido con $V = \{1, \dots, 7\}$ y con aristas E determinadas por $(1, 2), (1, 3), (3, 7), (3, 6), (2, 4)$ y $(2, 5)$ iniciando en el vértice (2) .
 - a) Escriba la lista de vértices en el orden en el que los visitaríamos en DFS.
 - b) Hay otro orden posible adicional al que escribió en el numeral anterior?
 - c) Cuántos órdenes DFS posibles hay para el grafo? Escribalos todos.
5. Sea G un grafo finito dirigido.
 - a) Complete rigurosamente la siguiente definición: Un orden topológico para G es una función...
 - b) Demuestre que si G admite un orden topológico f entonces no puede contener un ciclo dirigido.

6. Haga una implementación recursiva de `TopologicalSort`.
- a) Escriba su código en Python de la misma.
 - b) Qué orden topológico obtiene para el grafo con vértices s, v, w, t y lista de adyacencia $\{s : [v, w], v : [t], w : [t]\}$?
 - c) Qué hace su implementación si el input es el grafo con vértices s, v, w y lista de adyacencia $\{s : [v], v : [w], w : [s]\}$? Explique su respuesta de manera precisa.