

## TALLER DE GRAFOS

### 1. INTRODUCCION A LOS GRAFOS.

- Definición de grafos:

**Dirigidos:** Las aristas se representan con flechas, El nodo origen precede al nodo destino, El grado de un vértice se divide en grado de entrada y grado de salida.

**No dirigidos:** Las aristas representan relaciones simétricas, Cada arista se puede recorrer en ambas direcciones, El grado de un vértice es el número de aristas que lo tocan.

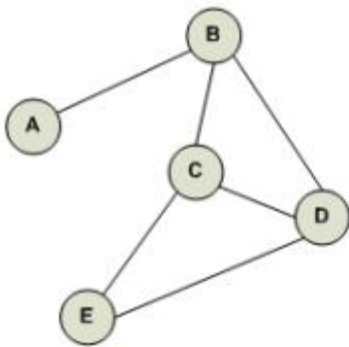


Fig 1. Grafo no dirigido

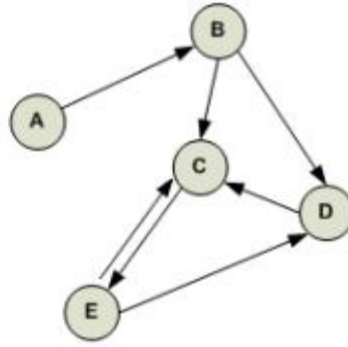
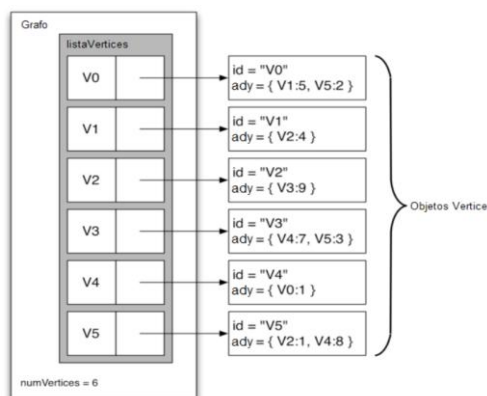


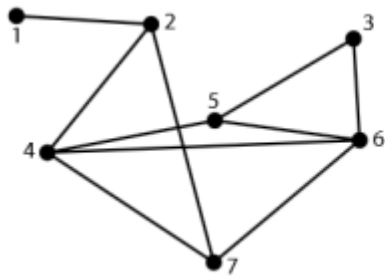
Fig 2. Grafo dirigido

- Representación con listas de adyacencia y matrices de adyacencia.

**Lista de adyacencia:** Una forma más eficiente, respecto al uso del espacio, de implementar un grafo conectado de forma rara es usar una lista de adyacencia. En una implementación de lista de adyacencia mantenemos una lista maestra de todos los vértices en el objeto Grafo y además cada objeto Vértice en el grafo mantiene una lista de los otros vértices a los que está conectado.



**Matriz de adyacencia:** La matriz de adyacencia es una matriz de  $n \times n$  donde  $n$  es la cantidad de nodos del grafo, que en la posición  $(i, j)$  tiene un 1 (o true) si hay una arista entre los nodos  $i$  y  $j$  y 0 (o false) si no.



$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}$$

- Casos de uso básicos en la vida real.

Detección de fraude.

Motores de recomendaciones.

Optimización de rutas.

Descubrimiento de patrones.

Gestión de conocimiento.

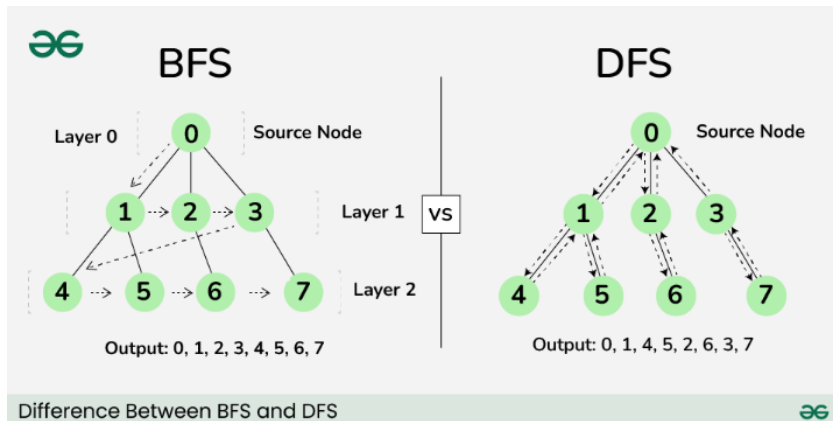
## 2. ELEMENTOS CLAVES PARA RESOLVER GRAFOS.

- Recorridos en grafos: BFS (anchura) y DFS (profundidad).

La búsqueda en anchura (BFS) y la búsqueda en profundidad (DFS) son dos algoritmos que se utilizan para recorrer grafos. Ambos son eficaces para distintos tipos de problemas.

**Búsqueda en anchura (BFS):** Se explora cada nivel del grafo a la vez, se recomienda para buscar el camino más corto en grafos no ponderados, se utiliza para la integridad y la optimalidad, Se utiliza para buscar el número mínimo de pasos entre dos nodos.

**Búsqueda en profundidad (DFS):** Se explora una rama del grafo lo más abajo posible antes de retroceder para explorar ramas alternativas, se utiliza para encontrar todas las rutas o soluciones posibles entre dos nodos, se utiliza para ahorrar memoria y evitar explorar nodos irrelevantes, se utiliza para detectar ciclos en un grafo.



### 3. Casos de Uso.

#### Caso de Uso 1: Ruta más corta entre dos puntos

Se desea encontrar la ruta más corta entre dos nodos en un grafo no ponderado (sin pesos en las aristas). El algoritmo de Búsqueda en Amplitud (BFS) es ideal porque explora los nodos nivel por nivel, garantizando que la primera vez que se llegue al destino será por la ruta más corta en términos de número de aristas.

#### Caso de Uso 2: Identificación de componentes conectados

En un grafo no dirigido, se busca identificar grupos de nodos que están conectados entre sí (componentes conectados). El algoritmo de Búsqueda en Profundidad (DFS) recorre todos los nodos accesibles desde un punto inicial, marcando cada componente conectado.

#### Caso de Uso 3: Detección de ciclos en un grafo dirigido

En un grafo dirigido, un ciclo ocurre si al explorar los nodos se encuentra un nodo ya visitado en la misma rama de exploración. Usamos DFS con un conjunto adicional para rastrear nodos en la pila de recursión.

#### Caso de Uso 4: Recomendación de amigos en redes sociales

En una red social representada como un grafo (nodos = personas, aristas = amistades), se busca recomendar amigos a un usuario basándose en amigos en común. Se explora el grafo para encontrar nodos conectados indirectamente (amigos de amigos).

### **Caso de Uso 5: Planificación de rutas de entrega**

En un grafo ponderado (nodos = ubicaciones, aristas = distancias), se busca la ruta más corta desde un punto de partida a todos los destinos. El algoritmo de **Dijkstra** calcula las distancias mínimas desde un nodo inicial.