UT8 Grafos NO Dirigidos

Definiciones

Grafo:
Un grafo G es una estructura formada por un conjunto de vértices V y un conjunto de aristas E,
donde cada arista es un par no ordenado de vértices.

Grafo No Conexo:
Un grafo es no conexo si existe al menos un par de vértices tal que no hay un camino entre ellos. En
otras palabras, el grafo se puede dividir en dos o más subgrafos disjuntos.

Subgrafo:
Un subgrafo H de un grafo G es un grafo cuyos vértices y aristas son subconjuntos de los vértices y
aristas de G.

> Ejemplo de un Subgrafo:
> Supongamos que tenemos el grafo G con los vértices {1, 2, 3, 4} y las aristas {{1, 2}, {2, 3}, {3, 4}}. Un subgrafo H podría ser el grafo con los vértices {2, 3} y la arista.

Grado de un Vértice:
El grado de un vértice en un grafo no dirigido es el número de aristas que inciden en él.

Ciclo:
Un ciclo es una secuencia de vértices VI, V2, ... , Vk tal que VI t,'k y cada par {Vi,

Vi+l} es una
arista en el grafo.

Metodos de representacion

Lista de Adyacencia

- Cada vértice tiene una lista de los vértices a los que está conectado.

```
1: [2, 3]
2: [1, 3, 4]
3: [1, 2, 4]
4: [2, 3]
```

```
1 - 2
| \ |
3 - 4
```

Matriz de Adyacencia

Una matriz n×nn \times nn×n donde nnn es el número de vértices. Si hay una arista entre el vértice i y el vértice j, la celda (i,j)(i, j)(i,j) es 1, de lo contrario es 0.

```
   1  2  3  4
1  0  1  1  0
2  1  0  1  1
3  1  1  0  1
4  0  1  1  0
```

```
1 - 2
| \ |
3 - 4
```

Arboles abarcadores de costo minimo

- Un árbol abarcador de costo mínimo (MST) es un subgrafo que conecta todos los vértices del grafo original con el menor costo posible sin formar ciclos. El costo de un MST es la suma de las aristas que lo componen.

La propiedad AAM

Un árbol abarcador mínimo tiene las siguientes propiedades:

- Conecta todos los vértices del grafo.
- Minimiza la suma de los pesos de las aristas.
- No contiene ciclos.

Algoritmo de Prim

El algoritmo de Prim construye el MST de un grafo de manera incremental, empezando por un vértice y agregando la arista más barata que conecte un vértice en el MST con uno fuera de él.

Pasos:

1. Inicializar un conjunto de vértices incluidos en el MST con un solo vértice inicial.
2. Repetir hasta que todos los vértices estén incluidos en el MST:
    - Seleccionar la arista de menor peso que conecta un vértice del MST con uno fuera de él.
    - Agregar el nuevo vértice al conjunto de vértices del MST.

```
Iniciar con un vértice         Seleccionar arista mínima
     (A)                              (A)
       \                             / |
        \                          /   |
         (B)     →             (B)---(D)
       /                     /   |
```

```
    /                      /    |
  (C)                  (C)
```

Algoritmo de Kruskal

El algoritmo de Kruskal también construye un MST pero ordenando primero todas las aristas por peso y luego agregando aristas al MST, asegurándose de no formar ciclos.

Pasos:

1. Ordenar todas las aristas del grafo por peso.
2. Inicializar un conjunto de componentes, uno por cada vértice.
3. Repetir hasta que el MST tenga $|V|-1$ aristas:
    - Seleccionar la arista de menor peso.
    - Si agregar esta arista no forma un ciclo, añadirla al MST y unir los componentes.

```
Ordenar aristas por peso        Unir componentes sin ciclos
     (A)-(B) 1                            (A)
     (B)-(D) 2                            /   \
     (A)-(C) 3                (B)        (B)   (D)
     (C)-(D) 4              /       →       /     \
     (B)-(C) 5            (C)            (C)---(D)
```

Recorridos

- Los recorridos en grafos son técnicas para visitar todos los vértices y aristas de un grafo. Los más comunes son la búsqueda en profundidad (DFS) y la búsqueda en amplitud (BFS).

Busqueda en profundidad

- DFS explora tan lejos como sea posible a lo largo de cada rama antes de retroceder. Utiliza una pila (ya sea explícita o la pila de llamadas recursivas).
- **Pasos:**
  1. Iniciar en un vértice v.
  2. Marcar v como visitado.
  3. Para cada vecino u de v, si u no ha sido visitado, realizar DFS desde u.

```
DFS desde A:
A -> B -> D -> C
 (A)
  |
 (B)
  |
 (D)
  |
 (C)
```

Busqueda en amplitud

- BFS explora todos los vecinos de un vértice antes de moverse a los vecinos de esos vecinos. Utiliza una cola.

Pasos:

1. Iniciar en un vértice v.
2. Marcar v como visitado y agregarlo a una cola.
3. Mientras la cola no esté vacía:
   - Sacar el vértice v de la cola.
   - Para cada vecino u de v, si u no ha sido visitado, marcar u como visitado y agregarlo a la cola.

```
BFS desde A:
A -> B -> C -> D
```

```
(A)
 |
(B)
 |
(C)
 |
(D)
```

Puntos de articulacion y componentes biconexos

Puntos de Articulación

Un punto de articulación es un vértice que, si se elimina junto con sus aristas, divide el grafo en componentes desconectados.

```
Grafo original:       Eliminar B:
(A)---(B)---(C)       (A)   (C)
 |     |     |        /       \
(D)---(E)---(F)     (D)     (F)

El vértice B es un punto de articulación.
```

Componentes Biconexos

Un componente biconexo es un subgrafo maximamente conectado, es decir, si se elimina cualquier vértice del subgrafo, este permanece conectado.

```
Grafo original:
(A)---(B)---(C)
 |     |     |
(D)---(E)---(F)

Componentes biconexos:
```

```
{A, D, E, B}
{B, C, F, E}
```