

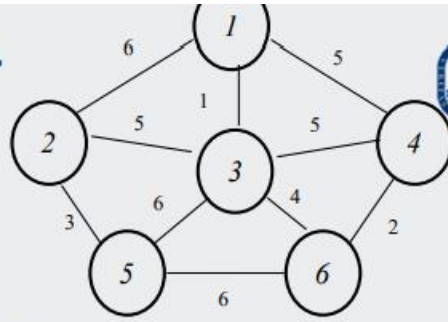
UT8 – Grafos no dirigidos

Propiedad AAM (Árbol Abarcador de costo Mínimo)



- Sea $G = (V, A)$ un grafo conexo con una función de costo definida para sus aristas. Sea U algún subconjunto propio del conjunto de vértices V .
 - Si (u,v) es una arista de **costo mínimo** tal que u pertenece a U y v pertenece a $V-U$, existe un AAM que incluye a (u,v) entre sus aristas.
- Dos algoritmos hacen uso de esta propiedad: Prim y Kruskal

Algoritmo de Prim.



Método TGRAFO.Prim (conjunto de aristas T);

U: conjunto de vértices;

u, v: vértice;

// el TGRAFO representado por un conjunto de vértices V y un conjunto de Aristas A

COMIENZO

T. Vaciar;

U.Agregar (1);

MIENTRAS U <> V **hacer**

elegir una arista (u,v) de costo mínimo

tal que u está en U y v está en V-U;

T.agregar (u,v);

U.agregar(v);

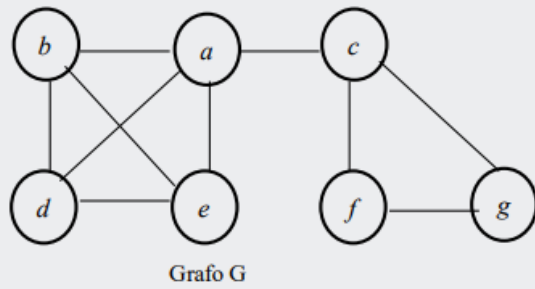
FIN MIENTRAS

FIN;

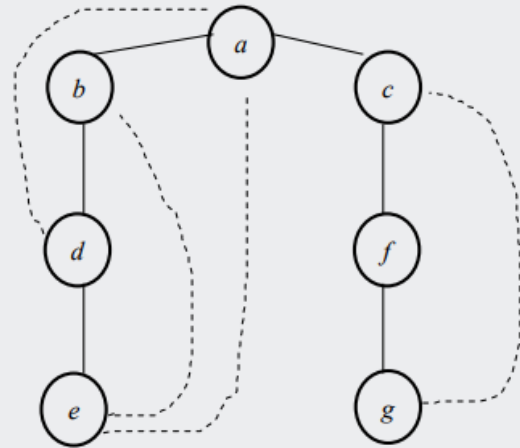
Algoritmo de Kruskal

- $G = (V, A)$, $V = \{1, 2, 3, \dots, n\}$ y una función de costo definida en las aristas de A
- Se empieza con un grafo $T = (V, \emptyset)$, constituido sólo por los vértices de G y sin aristas. Cada vértice es un componente conexo en sí mismo.
- Al avanzar, habrá siempre una colección de componentes conexos
- Para cada componente se seleccionarán las aristas que formen un árbol abarcador.
- Para construir componentes cada vez mayores, se agrega la arista de costo mínimo que conecte dos componentes distintos.
- La arista se descarta si conecta dos vértices que están en el mismo componente conexo, pues crearía un ciclo.
- Cuando todos los vértices están en un sólo componente, T es un árbol abarcador de costo mínimo para G .

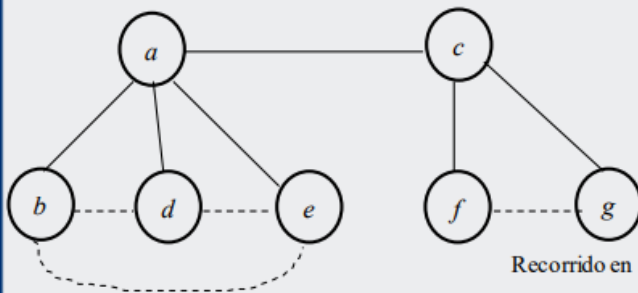
Búsqueda en Amplitud y Profundidad



Grafo G



Recorrido en profundidad



Recorrido en amplitud

Puntos de articulación

- El número **bajo** de un vértice v es el número más pequeño de ese nodo v o de cualquier otro w accesible desde él, siguiendo cero o más aristas de árbol hasta un descendiente x de v (x puede ser v), y después seguir una arista de retroceso (x, w) .
- Se calcula **bajo**(v) para todos los vértices v visitándolos en un recorrido en orden posterior. Cuando se procesa v , se ha calculado **bajo**(y) para todo hijo y de v . Entonces se toma **bajo**(v) como el mínimo de :
 1. **número_bp** de v .
 2. **número_bp** de z para cualquier vértice z para el cual haya una arista de retroceso (v, z) .
 3. **bajo**(y) para cualquier hijo y de v .

Puntos de articulación

