Árbol abarcador de costo mínimo

Colectivo Estructuras de Datos y Algoritmos

2021

1. Diga Verdadero (V) o Falso (F) según corresponda. Justifique su respuesta en cada caso.
   (a) Cuando a un grafo completo, no dirigido y ponderado se le quiere hallar el árbol abarcador de costo mínimo, el algoritmo más eficiente en tiempo es el de *Kruskal*.
   (b) Si la arista (u, v) es una arista de costo mínimo en un grafo, entonces (u, v) pertenece a algún **AACM**.
   (c) El conjunto de aristas $T = \{(u, v) \mid \exists (S, V - S)$ tal que (u, v) es una arista liviana que cruza $(S, V - S)\}$ forma un **AACM**.
   (d) Si la arista (u, v) está contenida en un **AACM** de un grafo , entonces en existe $(S, V - S)$ un corte para el cual la arista (u, v) cruza el corte y es liviana.
   (e) Sea T el árbol abarcador de costo mínimo del grafo conexo y ponderado $G =< V, E >$ y sean $a, b \in V$. El camino simple en T que conecta los vértices a y b se corresponde con el camino de costo mínimo en G que los conecta.
2. Demuestre los siguientes enunciados:
   (a) Sea G un subgrafo de G, y T' un **AACM** de G' , entonces $\exists T$ **AACM** de G tal que, $\forall e \in T$ que esté en G' también está en T'.
   (b) Una arista e pertenece a todo **AACM** de un grafo G si y solo si existe un corte de los vértices de G tal que e es la única arista liviana que cruza el corte.
3. Suponga que se define el costo de un árbol como el producto del costo de las aristas contenidas en el árbol en lugar de la suma. Implemente un algoritmo que dado un grafo $G =< V, E >$, cuyas aristas tienen costo positivo, permita determinar el **AACM** de G de forma eficiente. Justifique el orden y la correctitud de su algoritmo.
4. Sea $G =< V, E >$ un grafo no dirigido, ponderado y conexo. Se desea implementar un algoritmo que permita determinar si una arista $e \in E$ pertenece a algún **AACM** de G.
   (a) Demuestre que una arista (u, v) no pertenece a ningún **AACM** de G si y solo si en existe un camino de u a v formado por aristas cuyos costos son todos menores que el de (u, v)
   (b) Diseñe un algoritmo que en $O(|E|)$ dado un grafo $G =< V, E >$ y una arista $e \in E$ determine si e pertenece a algún **AACM** de G. Justifique el orden y la correctitud de su algoritmo.
5. Sea $G =< V, E >$ un grafo no dirigido, ponderado y conexo. Diseñe un algoritmo eficiente que permita determinar si una arista $e \in E$ dada, aparece necesariamente en todo **AACM** de G. Justifique el orden y la correctitud de su algoritmo.
6. Sea $G =< V, E >$ un grafo no dirigido, conexo y ponderado por una función $w : E \to \{1, 2\}$. Implemente un algoritmo que en $O(|E|)$ halle el árbol abarcador de costo mínimo de G. Justifique la correctitud y el orden de su algoritmo.
7. Sea G un grafo conexo y no dirigido y sea T un **AACM** de G. Suponga que se realiza una de las siguientes modificaciones a G:
   (a) Decrecer el peso de una de las aristas de G que no está en T.
   (b) Añadir un nuevo vértice a G con sus respectivas aristas.

   Proponga para cada caso un algoritmo que recalcule de forma eficiente el **AACM** de G luego de modificarlo. El costo de sus algoritmos debe ser asintóticamente menor que el costo de recalcular desde cero el **AACM**. Justifique la correctitud y el orden de los mismos