Clase 13-11

Andial

Noviembre 2024

1) Siguiendo las crónicas de la Batalla de los Anfibios, Francisco se dispone a analizar el impacto en el ancho de banda del ataque de los batracios. Se cuenta con los mismos datos que en el ejercicio 2: el conjunto de computadoras S, el de cables bidireccionales C, la computadora central s_c y el conjunto de amigxs $A \subseteq S$. Igual que antes, la red está dispuesta de tal forma que toda computadora puede alcanzar a la central mediante una secuencia de cables.

   Francisco averiguó el ancho de banda $w(c)$ de cada cable $c \in C$. Dada una secuencia de computadoras conectadas $s_1 s_2 \dots s_k$, se define el ancho de banda de la conexión como el mínimo de los anchos de banda $w(s_i s_{i+1})$ para $1 \leq i < k$. El ancho de banda entre dos computadoras s_i y s_j se denota como $bw(s_i, s_j)$, y se define como el máximo valor que puede tomar el ancho de banda considerando todas las secuencias de cables que conectan s_i y s_j.

   a) Dar un algoritmo con complejidad temporal $O(|C| \log |S|)$ que indique cuál es el menor ancho de banda entre las computadoras de lxs amigxs de Francisco y la computadora central. Es decir, el valor $\min_{s \in A}\{bw(s, s_c)\}$. Justificar su correctitud y complejidad.

   Como ya sabemos por el ejercicio anterior, los anfibios se disponen a romper uno de los cables esta misma noche, y Francisco busca entender cómo va a afectar este ataque en el ancho de banda de sus amigxs.

   b) Sea G un grafo conexo, T un AGM de G y e un eje de T. Sean T_1 y T_2 los dos subárboles disconexos que se obtienen al remover e del árbol T, y sea f alguna arista de menor peso que une a T_1 y a T_2 en G que es distinta a e. Probar que $T_1 \cup T_2 \cup \{f\}$ es un AGM de $G \setminus \{e\}$.

   c) Dar un algoritmo de complejidad temporal $O(|S||C|)$ que calcule el menor valor que puede alcanzar el ancho de banda de las computadoras de lxs amigxs de Francisco con la computadora central tras el ataque. Justificar su correctitud y complejidad.

   Criterio de aprobación: Por lo menos dos de los incisos tienen que estar bien resueltos.

2) Sea $G = (V, E)$ un grafo pesado con pesos positivos y $D \subseteq V$ un subconjunto de sus nodos, a los cuales llamaremos *depósitos*. Dado un nodo cualquiera v, decimos que su *distancia a los depósitos* es el mínimo entre las distancias de v a cada depósito $d \in D$, y lo notamos como $d_G(D, v)$. Es decir, $d_G(D, v) = \min\{d_G(w, v) : w \in D\}$.

   (a) Proponer un algoritmo que calcule la distancia a los depósitos de todos los nodos, justificando su diseño. Complejidad esperada: $O(n^2)$.

Decimos que una arista es *inútil* si **ningún camino mínimo** de los depósitos a los vértices usa esa arista.

b) Proponer un algoritmo que detecte las aristas inútiles, justificando su diseño. Complejidad esperada: $O(n^2)$.

Denotamos como $maxDist(G, D)$ a la máxima distancia de un nodo a los depósitos. Dicho formalmente, $maxDist(G, D) = \max\{d_G(D, v) : v \in V \setminus D\}$. Queremos agregar al grafo una arista e de un cierto costo positivo q de manera tal de minimizar $maxDist(G \cup \{e\})$[1]. Esta arista no puede estar conectada a ningún depósito.

c) Proponer un algoritmo que decida, dado el costo positivo q, qué arista se debe agregar al grafo para minimizar la máxima distancia a los depósitos. Aparte, indicar cuánto mejora esta máxima distancia. Complejidad esperada: $O(n^3)$.

Criterio de aprobación: Dos de los tres incisos deben estar bien resueltos.

3) En la Facultad de Ciencias Naturales y Exactas (FCNyE) se dictan M materias. Algunas deben cursarse en simultáneo, y por lo tanto al conjunto de todas las materias se lo particiona en k conjuntos disjuntos $G_1, \ldots, G_k$ con la intención de que les estudiantes cursen un grupo G_i por cuatrimestre.

Todas las materias deben tomar una evaluación en la última semana del cuatrimestre considerando las siguientes restricciones:

- Las evaluaciones de cada materia deben hacerse en los días en que se dicta la materia.
- No puede haber dos evaluaciones de un grupo G_i en un mismo día.
- Solo hay A aulas, por lo que cada día se pueden hacer a lo sumo A evaluaciones.

El Secretario Académico quiere saber si es factible armar una asignación de instancias de evaluación para que todas las materias puedan tomar sus evaluaciones en la última semana del cuatrimestre.

Nota: Suponer que la semana tienen 6 días hábiles, y que se cuenta con una función $DiasCursada(m)$ que devuelve, dada una materia m, una lista con los días de la semana en que se cursa la materia m[2].

Se pide:

(a) Modelar el problema de asignación como un problema de flujo.

(b) Dar una interpretación a cada unidad de flujo y cada restricción de capacidad.

(c) Justificar que el modelo es correcto. En particular, mostrar cómo se reconstruye una asignación válida en caso de que exista.

(d) Determinar la complejidad temporal del algoritmo en función del tamaño de la entrada. Asumir que para resolver el modelo de flujo se emplea el Algoritmo de Edmonds y Karp, y dar una cota ajustada. La entrada del algoritmo consiste en la partición $G_1, \ldots G_k$, el número A y la estructura de datos que responde las consultas de $DiasCursada(\cdot)$ de tamaño $O(M)$.

Criterio de aprobación: El modelado tiene que ser correcto, y se debe explicar claramente su diseño y su tamaño (en función de los parámetros de entrada).

[1] En un abuso de notación $G \cup \{e\}$ denota el grafo que se obtiene de agregar a G la arista e

[2] Se puede asumir que esta función devuelve la lista en $O(1)$.