

Estructuras de Datos y de la Información II

Examen final – 11 de junio 2004

Atención: el examen se entregará en tres bloques separados: preguntas 1 y 2, pregunta 3, y pregunta 4.

2 pt. 1. Las distancias respectivas entre las ciudades a, b, c, d, e y f son las siguientes:

a – b	10 km	b – c	40 km	c – d	30 km	d – e	10 km
a – c	30 km	b – d	50 km	c – e	20 km	d – f	40 km
a – e	40 km	b – f	10 km	c – f	20 km	e – f	60 km
a – f	20 km						

¿Cuántos kilómetros de cable telefónico se necesitarán como mínimo para tender una red que conecte a todas estas ciudades? Obtener la solución aplicando un algoritmo de grafos, mostrando paso a paso el estado de las estructuras de datos utilizadas en su ejecución.

2 pt. 2. a) Demostrar que la versión optimizada de Bellman-Ford computa las distancias mínimas (es decir, después de ejecutar BF (G, s) se cumple $d[u] = \delta(s, u) \forall u \in V[G]$). Indicación: para u accesible desde s, considerar un camino de coste mínimo de s a u y aplicar inducción.

b) Justificar que el siguiente algoritmo calcula igualmente las mismas distancias finales que Bellman-Ford. ¿Se construye necesariamente el mismo árbol? Demostrarlo o poner un contraejemplo.

Algoritmo (G, s)

 Inicializar (G, s)

 Visitar (s)

 Visitar (u)

 for $v \in \text{Ady}[u]$ do

 if $d[v] > d[u] + w(u, v)$ then

$d[v] \leftarrow d[u] + w(u, v)$

$\pi[v] \leftarrow u$

 Visitar (v)

Inicializar (G, s)

 for $v \in V[G]$ do

$d[v] \leftarrow \infty$

$\pi[v] \leftarrow \text{NIL}$

$d[s] \leftarrow 0$

c) Modificar el algoritmo del apartado anterior para que detecte ciclos de coste negativo en caso de haberlos.

3 pt. 3. Se tiene una base de datos de tamaño S Mb con N registros. Se dispone de R Mb de memoria RAM ($R \ll S$), y un único disco para lectura/escritura con un buffer de salida de B Mb.

a) Exponer brevemente dos razones por las que se querría ordenar la base de datos.

b) Justificar el coste de la ordenación del fichero en función del tiempo de acceso (ms) y el ratio de transferencia (Mb/ms) cuando se utilizan los siguientes algoritmos de ordenación:

- Ordenación por llave (Keysort), suponiendo que la tabla de claves cabe en RAM.
- Fusión simple (Mergesort).

c) Para los dos algoritmos de ordenación anteriores, estimar la diferencia de coste para ordenar el fichero cuando:

- El tamaño del fichero S (y el del fichero de índices) se incrementa en un factor 10.
- El tamaño de la memoria RAM se incrementa en un factor 10.
- El tamaño del buffer de salida se incrementa en un factor 10.
- Se utilizan dos discos, uno de lectura y otro de escritura.

(Continúa al dorso)