Auxiliar 1 - Cotas Inferiores

CC4102 - Diseño y Análisis de Algoritmos
Profesor: Pablo Barceló   Auxiliar: Jorge Bahamonde

20 de Marzo del 2015

1 Adversarios

Demuestre que:

1. Ordenar un arreglo de elementos toma al menos $\Omega(n\log n)$ comparaciones.
2. Determinar si un grafo no dirigido es o no conexo toma $\binom{n}{2}$ consultas del tipo "¿existe un arco entre los nodos u y v?", si la cantidad total de nodos es n.
3. Se necesitan al menos $2n-1$ comparaciones en el peor caso para mezclar dos listas crecientes de tamaño n en una única lista creciente.

2 Árboles de decisión

1. Demuestre que ordenar un arreglo de elementos toma al menos $\Omega(n\log n)$ comparaciones.
2. Dadas dos listas crecientes, de largos m y n, y suponiendo que $m < n$, existen dos formas obvias para mezclarlas, es decir, unirlas en una única lista creciente:
   - Tomar en cada "turno" el menor elemento de cada lista, lo que requiere $O(m+n)$ comparaciones.
   - Buscar cada elemento de la lista más corta en la más larga con búsqueda binaria, lo que requiere $O(m \log n)$ comparaciones.

   Se pide lo siguiente:

   - Usando árboles de decisión, encuentre una cota inferior al número de comparaciones necesarias para mezclar las dos listas.
   - Indique para qué relaciones entre m y n las técnicas descritas resultan ser óptimas.
   - Diseñe y analice un algoritmo de mezcla que alcance la cota inferior para toda relación entre m y n.
3. Se tiene un arreglo A ordenado de tamaño n. Suponga que sólo tiene disponible la función $\textsc{Test}(i,j,k,y)$, que dados $0 < i < j < k \leq n$ entrega una de las siguientes respuestas:

$$\begin{array}{cc} y < A[i] & A[j] < y < A[k] \\ y = A[i] & y = A[k] \\ A[i] < y < A[j] & y > A[k] \\ y = A[j] & \end{array}$$

   Use un árbol de decisión para mostrar una cota inferior sobre el número de llamados a $\textsc{Test}$ para encontrar la posición de un valor y en A (o retornar -1 si no se encuentra).