Examen Teórico 1

Análisis y Diseño de Algoritmos

22 de Junio del 2020

Ejercicio 1 (3 ptos). De un ejemplo de una permutación de los siete primeros números naturales (1...7) en donde el Quicksort hace la menor cantidad de comparaciones posibles. Justifique. ¿Cuántas comparaciones hace el Quicksort en su ejemplo?

Ejercicio 2 (4 ptos). Queremos codificar el archivo cuyo contenido es la siguiente cadena:

$$aaabbbbcccccccddddddddeeefffggggghhhhhhhh.$$

(a) Muestre el resultado final (el árbol y las codificaciones asociadas) luego de correr el algoritmo $O(n \lg n)$ de Huffman (no necesita mostrar pasos intermedios).

(b) Indique cuál es el peso del archivo codificado y compárelo con el peso de un archivo codificado usando una cantidad fija de bits para cada caracter. ¿Existe una mejora?

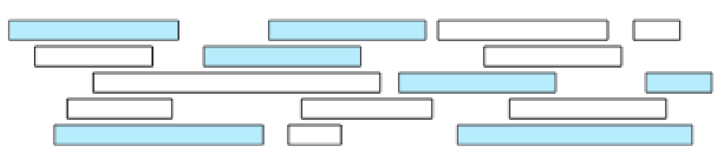
Ejercicio 3 (3 ptos). Respecto al problema de la mochila.

(a) Indique una instancia del problema de la mochila fraccionária, con al menos tres ítems, en donde la solución brindada por el algoritmo voraz no es una solución óptima para el caso entero.

(b) Para una instancia cualquiera del problema de mochila. ¿Existe alguna relación de desigualdad (mayor, menor, mayor o igual, menor o igual) entre los valores de las soluciones óptimas en los casos enteros y fraccionarios? Justifique.

Ejercicio 4 (6 ptos). Sea $\mathcal{I}$ un conjunto de n intervalos en la recta real. Decimos que un subconjunto $X \subseteq \mathcal{I}$ de intervalos *cubre* $\mathcal{I}$ si la unión de todos los intervalos en X es igual a la unión de todos los intervalos en $\mathcal{I}$.

Queremos diseñar un algoritmo voraz para encontrar el conjunto más pequeño de intervalos que *cubre* $\mathcal{I}$. Por ejemplo, en la siguiente figura, el número pedido es 7.



(a) Mencione cual es la elección voraz

(b) Escriba el pseudocódigo (no código) de su algoritmo voraz. Indique claramente qué recibe y qué devuelve su algoritmo. El algoritmo debe ser recursivo.

(c) Demuestre que su elección voraz es correcta. Enuncie a modo de lema y demuestre que su lema es correcto.

(d) Demuestre que su algoritmo tiene subestructura óptima. Enuncie a modo de lema y demuestre que su lema es correcto.

Deberá utilizar la siguiente notación.

Para los intervalos: $\mathcal{I} = \{[s_1, t_1], [s_2, t_2], \ldots, [s_n, t_n]\}$, donde $[s_i, t_i]$ es un intervalo con punta inicial s_i y punta final t_i. Para soluciones devueltas por el algoritmo o utilizadas en las demostraciones: variables X, Y, Y', X'.

Pista: ordene previamente sus intervalos según algún criterio (o considere que recibe los intervalos ordenados según ese criterio).

Ejercicio 5 (4 ptos). Sea $A[1..n]$ un arreglo de n números distintos. Si $i < j$ y $A[i] > A[j]$, entonces (i, j) es llamado una *inversión* de A. Suponga que los elementos de A son una permutación aleatoria uniforme de $1, 2, \ldots, n$. Calcule el número esperado de inversiones en A. Justifique usando el concepto de esperanza y variable aleatoria. (Pista: defina variables indicadoras para cada par de índices).