

Ejercicios en clase: División y conquista, Recurrencias

Análisis y Diseño de Algoritmos

17 de septiembre de 2021

Ejercicio 1. Ilustre la operación del MergeSort en el siguiente arreglo $A = \langle 3, 41, 52, 26, 38, 57, 9, 49 \rangle$.

Ejercicio 2. Considere la siguiente variación para insertionSort. Para ordenar el vector $A[1..n]$, ordenamos recursivamente el vector $A[1..n-1]$ y luego insertamos $A[n]$ en el arreglo ordenado $A[1..n-1]$. Escriba el pseudocódigo del algoritmo anterior. Escriba una recurrencia para el peor caso de este algoritmo. Resuelva la recurrencia.

Ejercicio 3. Considere el siguiente problema de búsqueda. Entrada: un arreglo ordenado $A[1..n]$, y un número v . Salida: Un índice i tal que $v = A[i]$ si v está en A y -1 si v no está en A . El algoritmo de búsqueda binaria para dicho problema encuentra el punto medio de A y lo compara con v , descartando la mitad de la secuencia y repitiendo este procedimiento recursivamente. Escriba el pseudocódigo del algoritmo anterior. Escriba una recurrencia para el peor caso de este algoritmo. Resuelva la recurrencia exactamente.

Ejercicio 4. Para cada uno de los siguientes ejercicios:

- Resuelva las recurrencias explícitamente olvidando el piso.
- Acote superior e inferiormente usando el ítem anterior. Concluya el orden Θ de la recurrencia.
- Compruebe usando inducción que la notación Θ encontrada en el ítem anterior es correcta. Debe hacerlo directamente de la definición, sin usar el resultado de los ítems anteriores.
- Compruebe usando teorema maestro en caso aplique (puede olvidar el piso para esta última comprobación).



En cada caso, suponga que $T(1) = 1$.

(a) $T(n) = 2T(\lfloor n/2 \rfloor) + n^2$

(b) $T(n) = 2T(n-1) + 3n - 2$

(c) $T(n) = 4T(\lfloor n/2 \rfloor) + n$

(d) $T(n) = 2T(\lfloor n/2 \rfloor) + n^3$

(e) $T(n) = 7T(\lfloor n/3 \rfloor) + n^2$