Ejercicios: Parcial

December 9, 2020

1 Análisis Asintótico

Ejercicio 1. Demostrar, usando las definiciones que

(a) $n^2 + 10n + 2 = O(n^2)$

(b) $\lceil n/3 \rceil = O(n)$

(c) $\lg n = O(\log_{10} n)$

(d) $n = O(2^n)$

(e) $\lg n$ no es $\Omega(n)$

(f) $n/100$ no es $O(1)$

(g) $n^2/2$ no es $O(n)$

Ejercicio 1. (7 ptos) Pruebe las siguientes afirmaciones.

(a) $2n^2 - 5n - 8 = \Omega(n^2)$

(b) $\sum_{k=1}^{n} k^{99} = \Theta(n^{100})$

Ejercicio 2. (7 ptos)

(*a*) (4 puntos) Resuelva la recurrencia por los métodos vistos en clase

$$T(n) = \begin{cases} 1 & n = 1 \\ T(n) = 8T(\lceil n/2 \rceil) + n & \text{caso contrário} \end{cases}$$

Puede suponer que es conocido que $T(n)$ es una función creciente.

(*b*) (3 puntos)

$$T(n) = \begin{cases} 1 & n = 1 \\ T(n) = 2T(n-1) + 3n - 20 & \text{caso contrário} \end{cases}$$

Pruebe por inducción que $T(n) = \Omega(2^n)$.

2 Divide-and-Conquer

Ejercicio 1. Decimos que un vector $A[1..n]$ es *unimodal* si existe un índice p, llamado *pico* tal que $A[1..p]$ es una secuencia creciente, y $A[p+1..n]$ es una secuencia decreciente. Diseñe un algoritmo de división y conquista que recibe un vector unimodal y encuentra el pico de A. Su algoritmo debe tener complejidad $\Theta(\lg n)$ en el peor caso. Escriba la recurrencia y resuélvala usando los métodos visto en clase. Verifique que su recurrencia es correcta usando el teorema maestro.

Ejercicio 3. (6 ptos)

Dado un arreglo $B[1..n]$, una k-rotación de B es un arreglo $A[1..n]$ tal que

$$A(k) = \begin{cases} B[i+k] & i+k \leq n \\ B[(i+k) \mod n] & \text{caso contrário} \end{cases}$$

Por ejemplo, si $B = [3, 6, 9, 10]$, una 2-rotación de B es $A = [9, 10, 3, 6]$.

Considere el siguiente problema.

Entrada: Una k-rotación $A[1..n]$ de un arreglo ordenado de elementos diferentes.

Salida: Un índice i tal que $A[i]$ es el elemento minimo del vector.

Observación: El número k es desconocido.

(a) (4 puntos) Diseñe un algoritmo de división y conquista que consuma tiempo $\Theta(\lg n)$ en el peor caso. Puede asumir que n es potencia de 2.

(b) (1 puntos) Escriba la recurrencia que define el peor caso en el tiempo de ejecución del algoritmo. Puede asumir que todas las constantes asociadas valen 1.

(c) (1 puntos) Resuelva la recurrencia anterior usando teorema maestro

Ejercicio 6. Considere el siguiente problema. Entrada: Un arreglo $A[1..n]$ de números enteros positivos ordenado de manera creciente. Salida: El mínimo número entero positivo que no está en A. Por ejemplo, si $B = [1, 2, 3, 4, 7, 8, 10, 12]$, el algoritmo debe devolver el valor 5.

Diseñe un algoritmo $\Theta(\lg n)$ en el peor caso. Escriba el pseudocódigo del algoritmo anterior. Escriba una recurrencia para el peor caso de este algoritmo. Resuelva la recurrencia. Verifique con teorema maestro. Verifique usando inducción.

3 Análisis Amortizado

Ejercicio 1. Suponga que en el problema de operaciones en pila incluimos una nueva operación MULTIPUSH, que pone $k \leq n$ elementos en la pila. ¿En esa situación, al hacer n operaciones, el costo amortizado de $O(1)$ por operación continúa valiendo? Suponga que ponemos la restricción de que el tamaño de la pila no puede sobrepasar n (el número de operaciones). ¿En esa situación el costo amortizado de $O(1)$ continúa valiendo?

Ejercicio 2. Muestre que si una operación DECREMENT es incluida en el problema del contador binario, n operaciones pueden costar $\Theta(nk)$.

Ejercicio 4. Suponga que hacemos una secuencia de n operaciones en una estructura de datos en la cual la i-ésima operación cuesta i si i es una potencia de 2, y cuesta 1 en caso contrário. Use análisis agregado para determinar el costo amortizado por operación.