



Primer examen opcional
FUNDAMENTOS DE ANÁLISIS Y DISEÑO DE ALGORITMOS
Grupo 51

Duración: 2 horas
Carlos Andres Delgado S, Ing *

06 de Mayo de 2015

Nota: Por favor marcar todas las hojas del examen con su nombre y código de estudiante. Recuerde colocar los procedimientos debido a que tienen un gran valor en la calificación.

1. Computación iterativa [39 puntos]

Se requiere un algoritmo que permita calcular una función cuya entrada es un número entero n y salida es $\sum_{i=0}^n (i^3 + \sum_{j=0}^{i+3} 2j)$

- [4 puntos.] Indique las salidas para las siguientes entradas: $n = \{0, 1, 3, 5, 7\}$
- [35 puntos.] Diseñe un algoritmo **utilizando ciclos** que solucione el problema
 - [6 puntos.] Escriba el algoritmo diseñado.
 - [11 puntos.] ¿Cual es la complejidad espacial y temporal de su algoritmo para una entrada de tamaño n en términos de $O(f(n))$?. Indique cuantas veces se ejecuta cada línea. Justifique su respuesta.
 - [4 puntos.] ¿Como puede representar los estados de su solución?. ¿Cual sería el estado inicial?
 - [6 puntos.] ¿Cual es la transición de estados de su algoritmo?.
 - [8 puntos.] ¿Cual es la invariante de ciclo de su algoritmo?.

2. Crecimiento de funciones y ecuaciones de recurrencia [16 puntos]

- [8 puntos] Suponiendo que $f(n)$ está definida demuestre que no existe una función $g(n)$ tal que cumpla $f(n) = o(g(n))$ y $f(n) = \omega(g(n))$.
- [8 puntos] Resuelva la ecuación $T(n) = 6T(\frac{n}{6}) + 4n^2$ con $T(1) = 0$ utilizando método del maestro.

3. Estructuras de datos [45 puntos]

3.1. Conceptos teóricos ([20 puntos])

Indicar cuales de las siguientes expresiones son verdaderas y falsas, en caso de ser falsas justifique su respuesta.

1. Estructura Pilas (Stack) y Colas (Queue)

- a) [4 puntos.] Una cola se puede considerar una estructura *FIFO* es decir, que el primer elemento encolado es el primero en ser desencolado.
- b) [4 puntos.] La complejidad de las operaciones **Push** y **Pop** en una pila es $\Theta(n \log(n))$

2. Estructura Listas

- a) [4 puntos.] En la lista **simplemente enlazada** la operación $LIST - INSERT(L, x)$ inserta x al inicio de la lista
- b) [4 puntos.] En una lista **doblemente enlazada** la complejidad de $LIST - DELETE(L, x)$ será $\mathcal{O}(1)$

3. Estructura Tablas Hash

* carlos.andres.delgado@correounivalle.edu.co

- a) [4 puntos.] Si se utiliza una buena función hash en una tabla hash con un número de **slots** m y un universo de llaves de tamaño k la complejidad de una **búsqueda** (*exitosa o no*) es $\Theta(n)$

3.2. Ejercicio práctico ([25 puntos])

Construya un árbol rojinegro insertando los siguientes elementos en este orden (13, 2, 3, 7, 9, 4, 5, -1, 7, 19, 18, 10). Muestre el procedimiento para agregar elemento por elemento.

Ayudas

Formulas de sumatorias

- $\sum_{k=1}^n c = cn$
- $\sum_{k=1}^n k = \frac{n(n+1)}{2}$
- $\sum_{k=1}^n k^2 = \frac{n(n+1)(2n+1)}{6}$
- $\sum_{k=0}^n ar^k = \frac{ar^{(n+1)} - a}{r-1}$ Si $r \neq 1$
- $\sum_{k=0}^n ar^k = (n+1)a$ Si $r = 1$

Formulas solución método del maestro

Recuerde la forma $T(n) = aT(\frac{n}{b}) + f(n)$

- Si $f(n) = O(n^{\log_b a - \epsilon})$ para algún $\epsilon > 0$ entonces $T(n) = \Theta(n^{\log_b a})$
- Si $f(n) = \Theta(n^{\log_b a})$ entonces $T(n) = \Theta(\log(n) * n^{\log_b a})$
- Si $f(n) = \Omega(n^{\log_b a + \epsilon})$ para algún $\epsilon > 0$, $c < 1$ y $af(\frac{n}{b}) \leq cf(n)$ entonces $T(n) = \Theta(f(n))$.

Inserción árboles rojinegros

- Caso 1: x(rojo) es un hijo de un padre rojo y el tío de x es rojo, se pintan de negro padre y tío de x, el abuelo de x queda entonces de rojo. x es ahora el abuelo de x
- Caso 2: x(rojo) es un hijo derecho de un padre rojo y el tío de x, y, es ahora negro. Se rota a la izquierda p[x]. x ahora es el padre de x

- Caso 3: x(rojo) es el hijo izquierdo de un padre rojo y el tío es negro. Se cambian los colores de p[x] y p[p[x]]. Se rota a la derecha p[x].

Figura 1: Rotaciones rojinegros

