



Segundo examen opcional

FUNDAMENTOS DE ANALISIS Y DISEÑO DE ALGORITMOS

Duración: 1.5 horas
Carlos Andres Delgado S, Ing^{*}
Junio de 2015

Importante: Muestre el proceso que realizó en cada punto, ya que el procedimiento tiene un gran valor en la calificación del parcial.

1. Ordenamiento [30 puntos]

1. (20 puntos) Aplique el algoritmo Heapsort para ordenar el arreglo $A = \{4, 5, 8, 1, 2, 10\}$. Muestre cada uno de los pasos utilizando una representación de montones.
2. (10 puntos) Explique brevemente un algoritmo que ordene en tiempo $O(n)$ números entre 0 y n .

2. Programación dinámica y voraz [70 puntos]

Para el cálculo de la función de Ackermann se utiliza la siguiente expresión:

$$A(n, m) = \begin{cases} n + 1 & \text{Si } m = 0 \\ A(m - 1, 1) & \text{Si } m > 0 \wedge n = 0 \\ A(m - 1, A(m, n - 1)) & \text{Si } m > 0 \wedge n > 0 \end{cases}$$

Pista: Si se utiliza una función recursiva, se pueden generar soluciones solapadas que lleven a cálculos repetidos.

1. (15 puntos) Escriba una función recursiva (en pseudocódigo) que permita cuyas entradas son n y m y su salida es un número entero (Solución ingenua)
2. (40 puntos) Escriba una función (en pseudocódigo) utilizando programación dinámica. Explique como caracterizó la estructura óptima y cómo se construye la solución óptima. Recuerde que en una solución dinámica debe evitar cálculos innecesarios.
3. (15 puntos) Compare la solución ingenua y la dinámica para el siguiente cálculo $A(3, 5)$ ¿Cual hace menos operaciones? ¿Justifique su respuesta?

^{*}carlos.andres.delgado@correounivalle.edu.co