

# Certamen Recuperativo

## Algoritmos y Complejidad

20 de enero de 2016



1. El algoritmo de Kruskal para hallar un árbol recubridor mínimo (*minimal spanning tree* en inglés) funciona como sigue: Dado un grafo, comienza con los vértices aislados y sucesivamente agrega un arco de mínimo costo que no crea ciclos en el grafo. Esboce un análisis del costo máximo en tiempo de este algoritmo en término del número de vértices  $v$  y arcos  $e$  del grafo.  
(20 puntos)
2. Demuestre formalmente que el algoritmo de la pregunta 1 es correcto.  
(30 puntos)
3. Un problema popular en entrevistas para trabajo es determinar rápidamente (tiempo mejor que  $O(n)$ ) en un arreglo ordenado que contiene elementos en pares, salvo un único elemento que no está en un par, cuál es el elemento que no se repite.  
**Pista:** Un rango de elementos repetidos debe tener largo par.  
(35 puntos)
4. Una QUEUE se implementa mediante dos STACK. La idea es que ENQUEUE pone un objeto en el tope del primer stack; DEQUEUE saca el elemento del tope del segundo, si éste está vacío transfiere todos los elementos del primero al segundo (vía POP y PUSH). Suponiendo que las operaciones sobre un STACK (PUSH, POP, EMPTY) todas tienen costo 1, use análisis amortizado para demostrar que las operaciones sobre QUEUE tienen costo amortizado  $O(1)$ .  
**Pista:** Una función potencial  $\Psi(D)$  es el número de elementos en el primer stack.  
(35 puntos)