

## Ejercicios sobre TADs arborescentes

### Estructuras de Datos y algoritmos 03/04, 25-11-03

Basados en los de [Weiss 93]:

1. Escribir algoritmos para calcular el mínimo y el máximo elemento de un árbol binario de búsqueda.
2. Demostrar por inducción que el máximo número de nodos en un árbol binario de altura  $h$  es  $2^{h+1}-1$ .
3. Demostrar por inducción que el número de punteros hijo que valen null en un árbol binario de  $n$  nodos es  $n+1$ .
4. En un árbol binario, un nodo completo es un nodo con dos hijos. Demostrar que el número de nodos completos es igual al número de hojas  $-1$ .
5. Dar el resultado de insertar 3, 1, 4, 6, 9, 2, 5, 7 en un árbol binario de búsqueda vacío. Dar el resultado de borrar la raíz.
6. Dar el resultado de insertar 2, 1, 4, 5, 9, 3, 6, 7 en un árbol AVL vacío.
7. Escribir procedimientos eficientes que dado un árbol binario (representado sólo por un puntero a la raíz) calculen: a) el número de nodos del árbol, b) el número de hojas del árbol, c) el número de nodos completos del árbol (ver ejercicio 4).
8. Escribir un procedimiento eficiente que dado un árbol binario de búsqueda y dos valores  $k_1$  y  $k_2$ , escriba todos los nodos del árbol con valores  $k$  que cumplen  $k_1 \leq k \leq k_2$ . Calcular el tiempo del algoritmo propuesto.
9. Escribir un programa que liste los nodos de un árbol binario por niveles, primero la raíz, luego los del nivel 1, luego los del 2, etc. Debe funcionar en tiempo lineal.
10. Dar el resultado de insertar 3, 1, 4, 5, 9, 2, 6, 8, 7, 0 en un árbol 2-3-4 vacío.
11. Dos árboles binarios son similares si están los dos vacíos o tienen subárboles izquierdo y derecho similares. Escribir una función que detecte si dos árboles binarios son similares. ¿Cuánto tarda?
12. Supongamos que queremos añadir a los árboles 2-3-4 la operación encontrar\_k\_ésimo, encontrar\_k\_ésimo( $T, i$ ) devuelve el valor del nodo de  $T$  que ocupa la posición  $i$ -ésima de menor a mayor. Supongamos que todos los nodos tienen valores distintos, modificar la representación para que esta operación tarde tiempo  $O(\log n)$  sin aumentar el tiempo de las demás operaciones.

Basados en los de [Franch 94]:

13. Especificar ecuacionalmente el modelo de árboles con punto de interés tanto binarios como generales, con un conjunto apropiado de operaciones.
14. Sean los árboles generales que contienen caracteres como valores de los nodos. Implementar un procedimiento que escriba todos los caminos que van de la raíz a las hojas.
15. ¿Es posible generar un árbol binario a partir únicamente de uno de sus recorridos preorden, inorden o postorden? ¿Y a partir de dos recorridos diferentes (examinar todos los pares posibles)? Dar los algoritmos para los casos que sean realizables.

16. Reconstruir un árbol binario con preorden: 2,1,4,7,8,9,3,6,5 e inorden: 7,4,9,8,1,2,6,5,3.
17. Reconstruir un árbol binario con inorden: 4,6,5,1,2,12,7,3,9,8,11,10 y postorden: 6,5,4,12,7,2,8,9,10,11,3,1
18. Construir un montículo (cola con prioridad) insertando sucesivamente los valores 64, 41, 10, 3, 9, 1 y 2. Borrar tres veces el mínimo y mostrar la evolución del árbol.
19. Dibujar todos los árboles de búsqueda posibles que contengan los naturales 1, 2, 3 y 4 (sin repeticiones). ¿Cuáles son equilibrados? En general, ¿cuántos árboles de búsqueda se pueden formar con n elementos diferentes?

Otros:

20. Se debe diseñar un algoritmo que lea una secuencia de palabras y determine el número de veces que aparece cada una de ellas en la secuencia. Para ello, se dispondrá de un árbol AVL inicialmente vacío. Cada palabra leída se buscará en el árbol; si se encuentra, se incrementará su contador; si no, se insertará en el árbol como nueva palabra (con el contador inicializado a 1). Una vez leído el texto, se deberán escribir en pantalla, por orden alfabético, las palabras leídas junto con el número de apariciones de cada una de ellas en la secuencia.
21. El tres en raya es un juego de dos jugadores con tres fichas cada uno que juegan sobre un tablero 3x3. Al principio el tablero está vacío. Los jugadores colocan alternativamente una ficha cada uno sobre una casilla libre del tablero. Cuando las seis fichas están sobre el tablero, el jugador al que le toca coge una de sus fichas y la coloca sobre una posición libre (distinta de la que estaba). Gana el jugador que consigue antes un "tres en raya", es decir, sus tres fichas alineadas en vertical, horizontal o una de las dos diagonales.

Se trata de implementar un algoritmo en el que un jugador juega contra la máquina al tres en raya, y la máquina utiliza la estrategia minimax para elegir su siguiente jugada. Al principio el programa debe preguntar al jugador si desea empezar, en caso contrario la máquina coloca la primera ficha.

Es importante notar que la mayoría de las configuraciones del tres en raya son "empates" si ambos jugadores utilizan la mejor estrategia. Por tanto a la hora de calcular la función de utilidad es suficiente explorar un número máximo de 5 niveles del árbol, si en esos cinco niveles no se demuestra que una configuración es ganadora o perdedora, entonces se considera empatadora.