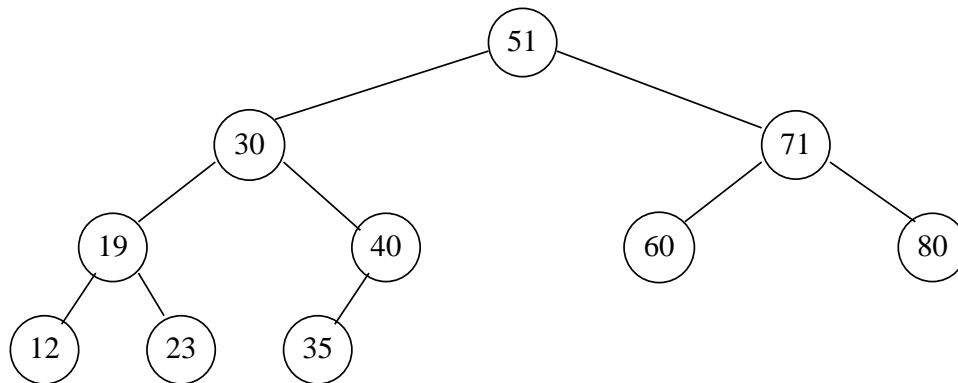


CC30A Algoritmos y Estructuras de Datos – Control 2

Profesor: Patricio Poblete

Mayo 29, 1998

1. Suponga que se tiene un *árbol de búsqueda binaria* con forma de *heap*. Por ejemplo,



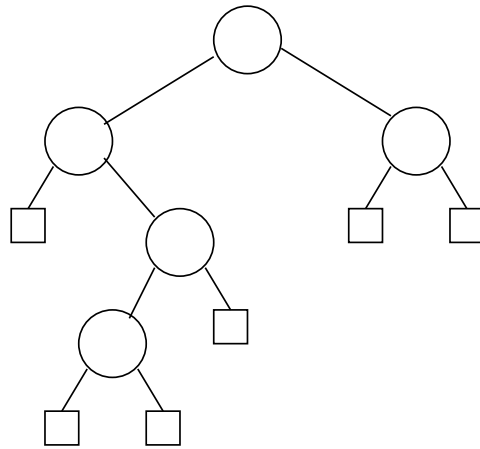
Almacenado este árbol en un arreglo, queda como sigue:

51	30	71	19	40	60	80	12	23	35
1	2	3	4	5	6	7	8	9	10

Programe una función de búsqueda de un elemento dado x , que opere sobre este arreglo. Esta función debe retornar el subíndice del casillero en donde está el elemento, o un cero si no está, y debe funcionar en tiempo logarítmico.

2. Un árbol puede representarse mediante un string de letras “x” y “o”, que se obtiene al recorrer el árbol en postorden, anotando una “o” cada vez que se encuentra un nodo interno, y una “x” cada vez que se encuentra una hoja.

Por ejemplo, el string “xxxoxoxxoo” representa al árbol



Escriba una función que reciba un string de este tipo, y retorne la altura del árbol representado por ese string. Para esto, considere al string como una “fórmula” escrita en notación polaca, la cual al ser evaluada entrega la altura.

- (a) En esa fórmula, ¿qué valor entrega una “x”?
 - (b) ¿Cuál es la operación representada por la “o”?
 - (c) Implemente la función pedida, utilizando un stack para la evaluación de la fórmula. Suponga que las operaciones básicas del stack ya están implementadas y disponibles para ser usadas por usted.
3. Escriba una función que reciba como parámetros un puntero a la raíz de un árbol de búsqueda binaria y dos valores `min` y `max` (suponga que `min < max`). La función debe imprimir, en orden ascendente, todas las llaves que están en el rango `min..max`.

Indicación: Use un recorrido del árbol como base, pero evite recorrer subárboles que se sabe que no contienen ninguna llave en el rango dado.

Tiempo: 2 horas

Entregar en hojas separadas

Con apuntes de clases