

UT4 - RAT

Árboles

Los árboles pueden definirse de forma recursiva y no recursiva, la definición no recursiva es la técnica más directa.

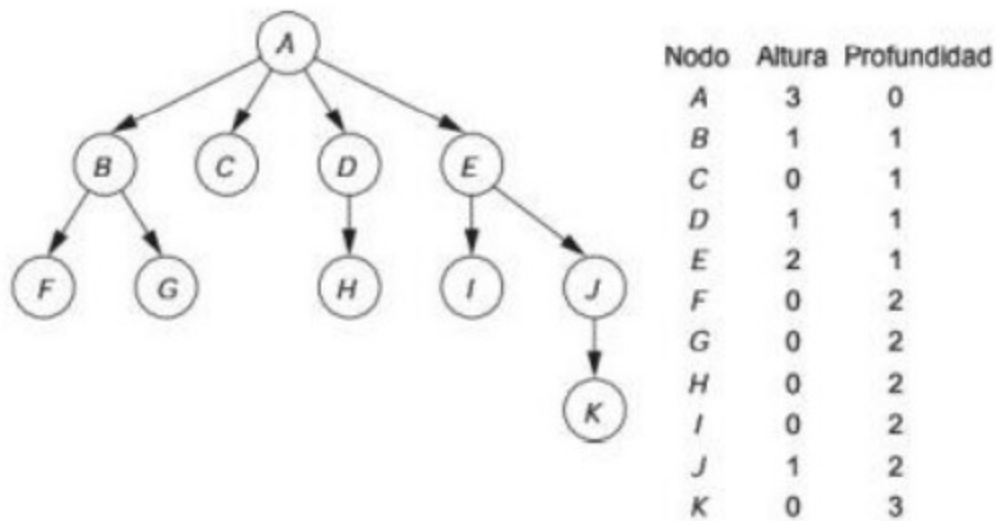
De forma **no recursiva**, un árbol está compuesto por un conjunto de nodos y un conjunto de aristas dirigidas que conectan parejas de nodos. Un árbol con raíz tiene las siguientes propiedades:

- Uno de los nodos es la raíz.
- Todo nodo c (menos la raíz) está conectado mediante una arista a exactamente un único otro nodo p . El nodo p es el padre de c y c es uno de los hijos de p .
- Existe un camino único que recorre el árbol desde la raíz hasta cada nodo. El número de aristas que hay que recorrer se denomina **longitud del camino**.

Un nodo que no tenga hijos se denomina **hoja**.

Un árbol con N nodos tendrá $N-1$ aristas, porque todo nodo (menos la raíz) tiene una arista entrante. La **profundidad** de un nodo es la longitud del camino desde la raíz hasta el mismo. La **altura** de un nodo es la longitud del camino que va desde el nodo hasta la hoja más profunda. Por lo tanto, la altura de un árbol es la altura de su raíz.

Si existe un camino desde el nodo u hasta el nodo v , entonces u es ancestro de v y v es descendiente de u . Si $u \neq v$ entonces u es un ancestro propio de v y v un descendiente propio de u . El **tamaño** de un nodo es el número de descendientes del mismo, incluyéndolo. Por ejemplo, en la imagen el tamaño de E es 3.



De forma **recursiva** un árbol o bien está vacío o está compuesto de una raíz y cero o más subárboles no vacíos, cuyas raíces están conectadas con la raíz del árbol mediante una arista (en algunos casos, especialmente en árboles binarios, se puede dar que algunos de los subárboles estén vacíos).

Método del primer hijo/segundo hermano

Como el número de hijos en un nodo varía enormemente, y no se conoce de antemano, no es factible incluir en la estructura de datos enlaces directos a los hijos (se desperdicia demasiado espacio). Por eso se implementa este método.

Se mantienen los hijos de cada nodo en una lista enlazada de nodos de árbol en la que cada nodo mantiene dos enlaces, uno a su hijo situado más a la izquierda, y otro al hermano situado a su derecha.

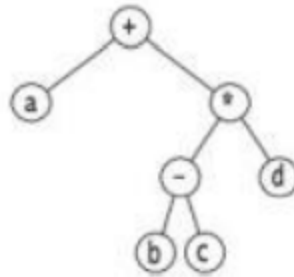
Sistemas de archivos

Uno de los usos más populares de los árboles es la estructura de directorios en muchos sistemas operativos. El asterisco situado junto al nodo indica que es un directorio. Si se siguen las aristas se forma una ruta (cada arista indica un /) Si la ruta comienza en la raíz se forma una ruta absoluta (completo) y sino una ruta relativa (relativa a su directorio).

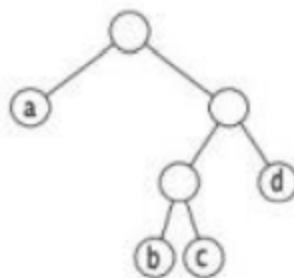
Árboles binarios

Un árbol binario es un árbol en el que ningún nodo puede tener más de dos hijos (izquierdo y derecho). Recursivamente, un árbol binario o está vacío o está compuesto de una raíz, un árbol izquierdo y uno derecho. Los árboles izquierdo y derecho pueden estar vacíos.

Árbol de expresión las hojas de expresión son operandos, los nodos restantes contienen operadores.



Árbol de codificación de Huffman se utiliza para implementar un algoritmo de comprensión de datos simple, pero efectivo. Cada símbolo del alfabeto se almacena en una hoja. Su código se obtiene siguiendo el camino que va hasta esa hoja desde la raíz. Cada enlace izquierdo corresponde a un 0 y cada enlace derecho a un 1. Así en el siguiente ejemplo, el código de b sería 100.



Recorrido en preorden

El trabajo del nodo se lleva a cabo antes de procesar a sus hijos. Es un algoritmo compacto, y resulta eficiente porque requiere un tiempo constante por cada nodo.

Un nodo se declara como visitado después de haber sido extraído por primera vez.

Recorrido en postorden

El trabajo de un nodo se lleva a cabo después de evaluar a sus hijos. También requiere un tiempo constante por nodo.

Se implementa utilizando una pila para almacenar el estado actual. La cima de la pila representará el nodo que estamos visitando en algún instante dentro del recorrido en postorden.

La pila contiene nodos que hemos recorrido pero aún no hemos completado. Cuando se introduce un nodo en la pila, el contador será 1, 2 o 3, de acuerdo a lo siguiente:

1. Si estamos a punto de procesar el subárbol izquierdo del nodo.
2. Si estamos a punto de procesar el subárbol derecho del nodo.
3. Si estamos a punto de procesar el propio nodo.

Recorrido en orden (in order)

Primero se procesa el hijo izquierdo, luego el nodo actual y por último el derecho.

Un nodo se declara como visitado después de haber sido extraído una segunda vez.

```

1 // Imprimir árbol cuya raíz está en nodo actual usando recorrido en preorden.
2 public void printPreOrder( )
3 {
4     System.out.println( element ); // Nodo
5     if( left != null )
6         left.printPreOrder( );      // Izquierda
7     if( right != null )
8         right.printPreOrder( );     // Derecha
9 }
10
11 // Impr. árbol cuya raíz está en nodo actual usando recorrido en postorden
12 public void printPostOrder( )
13 {
14     if( left != null )              // Izquierda
15         left.printPostOrder( );
16     if( right != null )             // Derecha
17         right.printPostOrder( );
18     System.out.println( element ); // Nodo
19 }
20
21 // Imprimir árbol cuya raíz está en nodo actual usando recorrido en orden.
22 public void printInOrder( )
23 {
24     if( left != null )              // Izquierda
25         left.printInOrder( );
26     System.out.println( element ); // Nodo
27     if( right != null )
28         right.printInOrder( );      // Derecha
29 }

```

Árboles binarios de búsqueda

En un árbol de búsqueda binaria, para todo nodo X, todas las claves contenidas en el árbol izquierdo de X tienen valores inferiores al de las claves de X, y todas las claves contenidas en el árbol derecho de X tienen valores superiores al de la clave de X.

La operación más costosa es la de eliminación. Una vez que encontramos el nodo que queremos eliminar hay que tener en cuenta varias cosas, por ejemplo, si el nodo no es una hoja, se debe reasociar el árbol manteniendo la prioridad de orden que hace de él un árbol de búsqueda binaria.

Árboles AVL

ES un arbol de búsqueda binaria con la condición adicional de equilibrio de que, para cualquier nodo del árbol, las alturas de los subárboles izquierdo y derecho puede diferir como máximo en 1.