Matemáticas discretas II

Árboles

carlos.andres.delgado@correounivalle.edu.co

Carlos Andrés Delgado S.

Facultad de Ingeniería. Universidad del Valle

Marzo de 2017

Contenido

Contenido

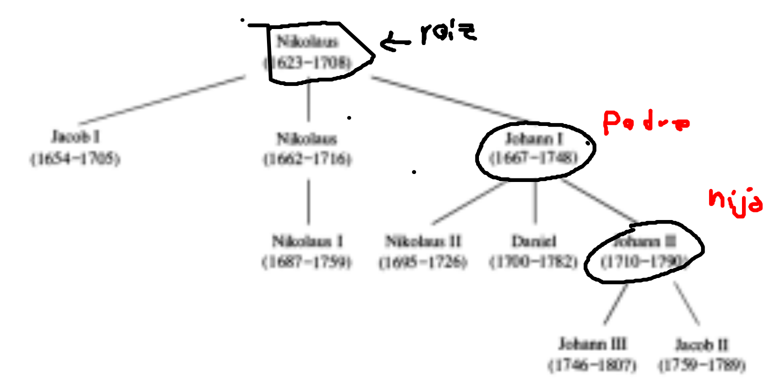
Introducción a los árboles

Introducción

Un árbol es un grafo conexo que no contiene circuitos simples. Los árboles son utilizados en un gran número de problemas computaciones, como es el caso de algoritmos de codificación, programación dinámica, entre otros.

Definición 1

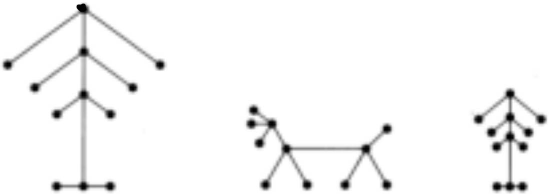
Un árbol no tiene ningún circuito simple. Esto quiere decir que no puede contener aristas múltiples ni ciclos.



Introducción a los árboles

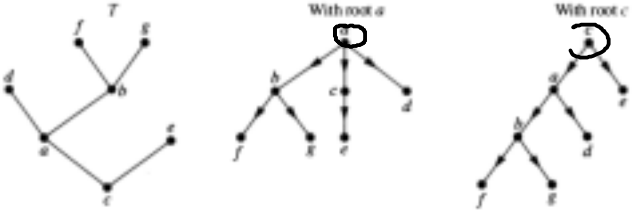
Teorema 1

Un grafo no dirigido es un árbol si y sólo si hay un único camino entre cada par de .



Definición 2

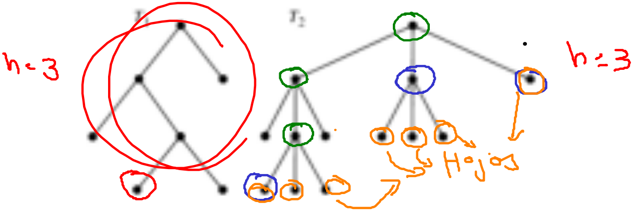
Un árbol con raíz es un árbol con un vértice que ha sido designado como raíz y cada arista se puede acceder desde un camino directo desde la raíz.



Introducción a los árboles

Definición 3

Un árbol con raíz es llamado m-ario si cada vértice interno no tiene más de m hijos. Un árbol es llamado un árbol m-ario completo si cada vértice interno tiene exactamente m hijos. Un árbol m-ario con $m = 2$ es llamado árbol binario.



Introducción a los árboles

Definición 4

La terminología de los árboles tiene orígenes botánicos y genealógicos. Suponga que T es un árbol con raíz.

- El **padre** de v es el único vertice u, tal que hay una arista dirigida de u a v
- El caso contrario anterior, se dice v es **hijo** de u
- Los vértices con el mismo padre son llamados **hermanos**
- Los **antecesores** de cualquier vértices, son el camino desde la **raiz** hasta el vértices, pero excluyéndolo a él
- Los **descendientes** de un vértice son todos aquellos que tienen a v como antecesor

Introducción a los árboles

Definición 5

- Un vértice es llamado **hoja** si no tiene hijos
- Los vértices ~~de los~~ con hijos son llamados **vértices internos** a

nodas

Definición 6

El **nivel de un vértice** es la longitud del único camino desde la raíz que hasta él. El nivel de la raiz es 0

Introducción a los árboles

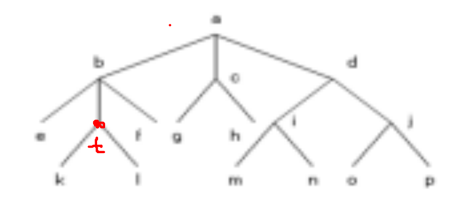
Definición 7

La **altura** de un árbol es la **longitud del camino más largo** desde la raíz hasta cualquier vértice

Definición 8

Un árbol de altura (h), está **equilibrado** o **balancedo** si todas sus hojas están en los niveles h o $h - 1$

Ejemplo de árbol equilibrado

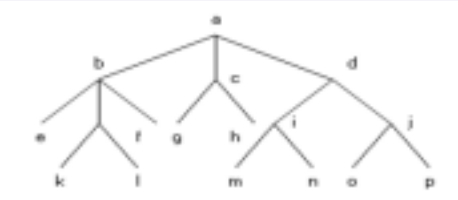


1. El árbol esta **equilibrado**
2. Su **altura** es 3
3. j es el **padre** de p
4. Los **antecesores** de n son $\{i, d, a\}$
5. los **descendientes** de d son $\{i, j, m, n, o, p\}$
6. Los vértices $\{e, k, l, f, g, h, m, n, o, p\}$ son **hojas**
7. Los vértices $\{a, b, c, d, i, j\}$ son **vértices internos**

Introducción a los árboles

Definición 9

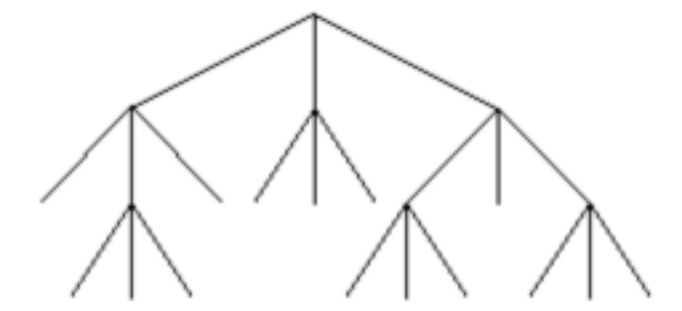
Un árbol es **m-ario** si todos los vértices tienen máximo m hijos. Un árbol m-ario con $m = 2$ es llamado á**rbol binario.**



Introducción a los árboles

Definición 10

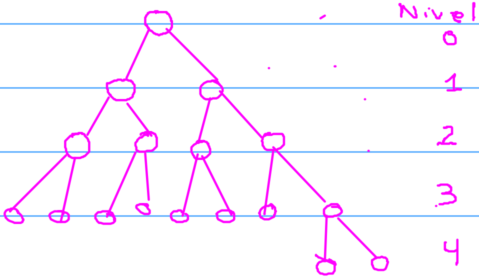
Un **árbol m-ario completo** es aquel donde los vértices internos tienen exactamente *m* hijos y es equilibrado.



Dibujar el arbol m-ario completa

$h = 4$

$m = 2$



Introducción a los árboles

Propiedades de los árboles

Un árbol con n vértices tiene $n-1$ aristas.

Demostración

Paso base: Con $n = 1$ se tiene 0 aristas.
Paso inductivo: Si se supone que para n existen $n-1$ aristas, ahora miramos para $n+1$, para conectar el nuevo vértice se necesita una nueva arista

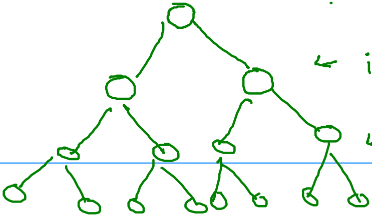
Introducción a los árboles

Propiedades de los árboles

Un árbol m-ario completo con *i* vértices internos contiene n = m*i + 1 vértices.

Demostración

Puesto que el número de nodos internos es *i* y cada uno tiene *m* hijos diferentes de la raiz.x



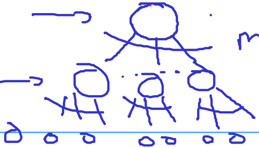
$i = 1, \quad m * i + 1$

$2 * 1 = 3$

$i = 3 \quad 2 * 3 + 1 = 7$

$i = 7 \quad 2 * 7 + 1 = 15$

$m^0 \to 1 \to$



m hijos

$m \to$ m

$m^2 \to$

m^3 ______

$m+1$

$(m+1)\times m$

$m^2+m \quad +1$

$(m^2+m+1)\times m$

m^3+m^2+m

Introducción a los árboles

Propiedades de los árboles

Un árbol m-ario de altura h, tiene máximo m^h hojas.

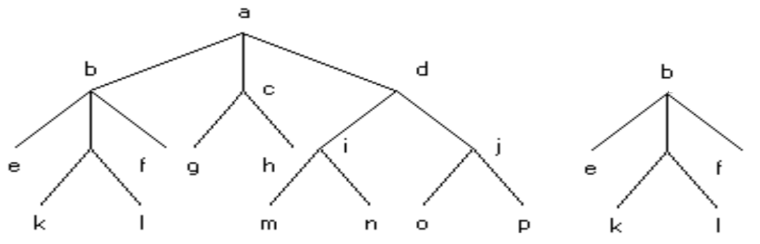
Demostración

- **Paso base** Con $h = 1$ se tiene m hijos
- **Paso inductivo** Tomando h cualquier se tienen m^h hijos. Para $h + 1$ cada hijo tiene m hijos, por lo tanto, se tienen en total $m * m^h = m^{h+1}$ hijos.

Introducción a los árboles

Propiedades de los árboles

Un **subárbol** es un árbol que se obtiene al tomar uno de los nodos internos de un árbol como raíz.



Contenido

Recorridos en árboles

Definición

Los árboles con raíz se utilizan frecuentemente para almacenar información. Existen algoritmos de recorrido para visitar cada uno de los vértices para acceder a los datos. Los algoritmos más conocidos de recorrido de árboles son:

- Recorrido en preorden
- Recorrido en inorden
- Recorrido en postorden

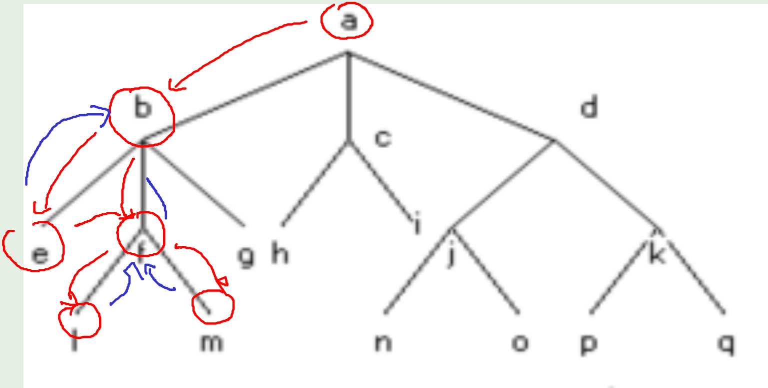
Recorridos en árboles

Recorrido en preorden

Para realizar el recorrido en preorden:

1. Visite la raíz. muestre la raíz
2. Visite los subárboles de izquierda a derecha (repita el procedimiento hasta llegar a las hojas)

Recorrido en preorden



El recorrido preorden es $a, b, e, f, l, m, g, c, h, i, d, j, n, o, k, p, q$

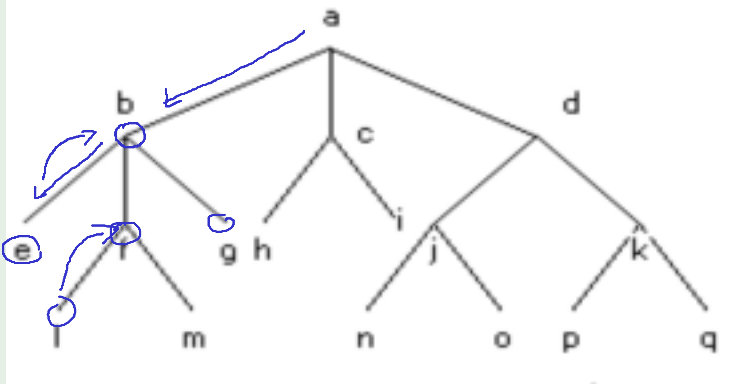
Recorridos en árboles

Recorrido en inorden

Para realizar el recorrido en inorden:

1. Visite el sub-arbol más izquierdo (primer hijo de izquierda a la derecha)
2. Visite la raíz. muestre la raíz
3. Visite los subárboles restantes de izquierda a derecha (repita el procedimiento hasta llegar a las hojas)

Recorrido en inorden



El recorrido en inorden es:

$e, b, l, f, m, g, a, h, c, i, n, j, o, d, p, k, q$

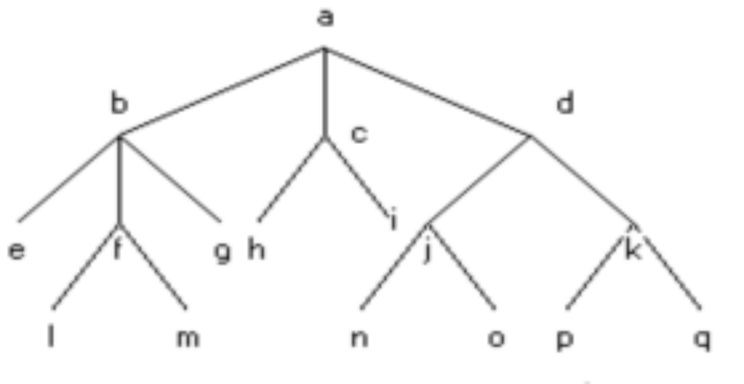
Recorridos en árboles

Recorrido en postorden

Para realizar el recorrido en postorden:

1. Visite el sub-arbol más izquierdo (primer hijo de izquierda a la derecha)
2. Visite los subárboles restantes de izquierda a derecha (repita el procedimiento hasta llegar a las hojas)
3. Visite la raíz. muestre la raíz

Recorrido en postorden



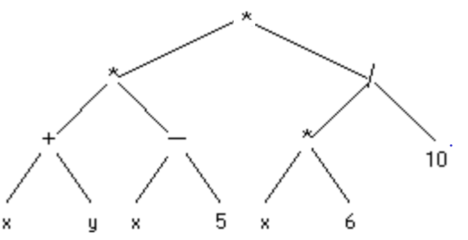
El recorrido en postorden es:

$e, l, m, f, g, b, h, i, c, n, o, j, p, q, k, d, a$

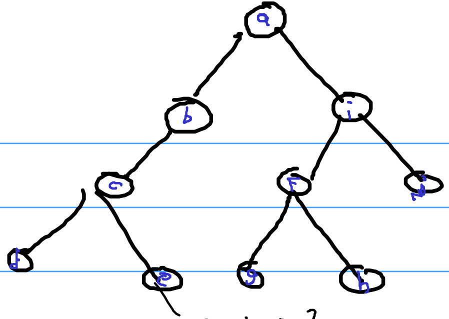
Recorridos en árboles

Expresiones aritméticas

Las expresiones matemáticas pueden ser representadas usando árboles:



- En preorden: (* (* (+ x y) (- x 5)) (/ (* x 6) 10))
- En inorden: ((x + y) * (x - 5)) * ((x * 6) / 10))
- En postorden: (((x y +) (x 5 -) *) ((x 6 *) 10 /) *)



Preorden= { a, b, c, d, e, i, F, g, h, j }

Inorden = { d, c, e, b, a, g, F, h, i, j }

Postorden = { d, e, c, b, g, h, F, j, i, a }