Background pattern

Description automatically generated with medium confidence **Universidad Abierta para Adultos**

**(UAPA)**



**Presentado por: Matrícula:**

Ana María Cepeda 100023687

Elibeth Michelle Tapia Cruz 100032092

Cesar Stalin Peña 100031571

Ángel Morezux 100030253

Roderi Daniel Peña 100026542

Carlos Guzmán 100020074

**Materia:**

Estructura de datos

**Asignación:**

No.3

Background pattern

Description automatically generated with low confidence **Facilitador:**

Maikel Aparicio

**Fecha:**

Marzo 2022

**Árboles binarios de búsqueda.**

Los árboles son estructuras de datos muy similares a las listas doblemente enlazadas, en el sentido que tienen punteros que apuntan a otros elementos, pero no tienen una estructura lógica de tipo lineal o secuencial como aquellas, sino ramificada. Tienen aspecto de árbol, de ahí su nombre.

Diagram

Description automatically generatedUn árbol es una estructura de datos no lineal en la que cada nodo puede apuntar a uno o varios nodos. También se suele dar una definición recursiva: un árbol es una estructura compuesta por un dato y varios árboles. Esto son definiciones simples. Una representación gráfica de los árboles se puede visualizar en la figura que se presenta debajo:

* Un Árbol consiste en un nodo (r, denominado nodo raíz) y una lista o conjunto de subárboles (A 1, A 2, .. A k).
* Si el orden de los subárboles importa, entonces forman una lista, y se denomina árbol ordenado (por defecto un árbol se supone que es ordenado). En caso contrario los subárboles forman un conjunto, y se denomina árbol no ordenado.
* Se definen como nodos hijos de r a los nodos raíces de los subárboles A 1, A 2, .. A k.
* Si b es un nodo hijo de a entonces a es el nodo padre de b.
* Un nodo puede tener cero o más hijos, y uno o ningún padre. El único nodo que no tiene padre es el nodo raíz del árbol.
* Un nodo sin hijos se denomina nodo hoja o externo. En caso contrario se denomina nodo interno.

Se define un camino en un árbol como cualquier secuencia de nodos del árbol, n 1 … n p, que cumpla que cada nodo es padre del siguiente en la secuencia (es decir, que ni es el padre de ni+1). La longitud del camino se define como el número de nodos de la secuencia menos uno (p-1).

* Los descendientes de un nodo ( c en el diagrama) son aquellos nodos accesibles por un camino que comience en el nodo.

A picture containing clipart

Description automatically generated

Los ascendientes de un nodo ( f en el diagrama) son los nodos del camino que va desde la raíz a él.

A picture containing clipart

Description automatically generated

**Altura.**

* Se define la altura de un nodo en un árbol como la longitud del camino más largo que comienza en el nodo y termina en una hoja. La altura de un nodo hoja es cero (0).

La altura de un nodo es igual a la mayor altura de sus hijos + 1.

La altura de un árbol se define como la altura de la raíz.

* La altura de un árbol determina la eficiencia de la mayoría de las operaciones definidas sobre árboles.

A picture containing clipart, clock

Description automatically generated

**Profundidad.**

* Se define la profundidad de un nodo en un árbol como la longitud del camino (único) que comienza en la raíz y termina en el nodo. También se denomina nivel.
* La profundidad de la raíz es 0.
* La profundidad de un nodo es igual a la profundidad de su padre + 1.

**Características.**

* Todo árbol tiene un único nodo raíz.
* Todo nodo, excepto la raíz está conectada por medio de un puntero o arista a un único nodo, conocido como nodo padre, que le(s) antecede.
* Todo nodo que no tiene más ramificaciones se le conoce como nodo terminal u hoja.
* Hay un único camino desde la raíz a cada nodo. El número de punteros que atraviesa es la longitud del camino.
* Padre es el antecesor inmediato de un nodo
* Hijo, es cualquiera de sus descendientes inmediatos.
* Hermano de un nodo, es otro nodo con el mismo padre.
* El grado de un nodo cualquiera, es el número de descendientes directos que tenga.
* Grado de un Árbol, es el máximo grado de todos los nodos.
* Altura de un árbol, es el máximo número de niveles de todos los nodos del árbol.
* Nivel, es el número de punteros o aristas o arcos que deben ser recorridos para llegar a un determinado nodo.

**Árboles binarios**.

A los árboles ordenados de grado dos se les conoce como arboles binarios ya que cada nodo del árbol no tendrá más de dos descendientes directos. Las aplicaciones de los árboles binarios son muy variadas ya que se les puede utilizar para representar una estructura en la cual es posible tomar decisiones con dos opciones en distintos puntos.

La representación gráfica del árbol binario es la siguiente:

Diagram

Description automatically generated

* Representación en memoria.

Hay dos formas tradicionales de representar un árbol binario en memoria:

* Por medio de datos tipo punteros también conocidos como variable dinámicas o listas.
* Por medio de arreglos.

Sin embargo, la más utilizada es la primera, puesto que es la más natural para tratar este tipo de estructura.

Los nodos del árbol binario serán representados como registro que contendrán como mínimo tres campos. En un campo se almacenará la información del nodo. Los dos restantes se utilizarán para apuntar al subárbol izquierdo y derecho del subárbol en cuestión.

**Clasificación de árboles binarios.**

Existen cuatro tipos de árbol binario.

* Distinto.
* Similares.
* Equivalentes.
* Completos.

**Distinto.**

Se dice que dos árboles binarios son distintos cuando sus estructuras son diferentes.

Ejemplo:

Diagram

Description automatically generated

**Similares.**

Dos árboles binarios son similares cuando sus estructuras son idénticas, pero la información que contienen sus nodos es diferente.

Ejemplo:

**Diagram

Description automatically generated**

**Equivalentes.**

Son aquellos árboles que son similares y que además los nodos contienen la misma información.

Ejemplo:

Diagram

Description automatically generated

**Completos.**

Son aquellos árboles en los que todos sus nodos excepto los del último nivel, tiene dos hijos; el subárbol izquierdo y el subárbol derecho.

**Recorrido de un árbol binario.**

Hay tres manera de recorrer un árbol: en inorden, preorden y postorden. Cada una de ellas tiene una secuencia distinta para analizar el árbol como se puede ver a continuación:

**Inorden.**

* Recorrer el subárbol izquierdo en inorden.
* Examinar la raíz.
* Recorrer el subárbol derecho en inorden.

**Preorden.**

* Examinar la raíz.
* Recorrer el subárbol izquierdo en preorden.
* Recorrer el subárbol derecho en preorden.

**Postorden.**

* Recorrer el subárbol izquierdo en postorden.
* Recorrer el subárbol derecho en postorden.
* Examinar la raíz.

Ejemplo de los diferentes recorridos en un árbol binario.

Diagram, shape

Description automatically generated

Diagram

Description automatically generated

**Inorden:** GDBHEIACJKF

**Preorden:** ABDGEHICFJK

**Postorden:** GDHIEBKJFCA

**Arboles binarios de búsqueda.**

Un árbol de búsqueda binaria es una estructura apropiada para muchas de las aplicaciones que se han discutido anteriormente con listas. La ventaja especial de utilizar un árbol es que se facilita la búsqueda.

Un árbol binario de búsqueda es aquel en el que el hijo de la izquierda (si existe) de cualquier nodo contiene un valor más pequeño que el nodo padre, y el hijo de la derecha (si existe) contiene un valor más grande que el nodo padre.

Ejemplo de árbol binario de búsqueda es el siguiente:

Diagram

Description automatically generated

**Código fuente, repositorio GIT. Url del repositorio.**