02/10/2019

Estructura de datos jerárquicas

Un árbol es una estructura de datos usada en programación, combina las ventajas de los arreglos ordenados y la lista enlazada. (se manipula rápido y se busca rápido).[son una especialización de lo grafos] características: Son pequeños arriba y van aumentando su tamaño.

Contiene sub Árboles.

Solo existe una ruta entre 2 pares de nodos.

Se utilizan como sistemas de archivo.

En las bases de datos la información se almacena como árboles B.

Algoritmos de compresión.

Compiladores usan árbol de expresión.

El nodo que está al tope del árbol es la raíz y no puede tener más de 1.

Un nodo sólo puede tener un padre.

Un nodo puede tener de 0...n líneas.

Cuando un nodo no tiene hijos es un nodo hoja.

Los Nodos que tienen en mismo padre son nodos hermanos.

largo de la LA RUTA es la cantidad de aristas que hay para llegar al nodo.

La profundidad de un nodo es la cantidad de aristas que hay desde el nodo hasta la raíz.

Altura de un nodo es el largo desde el nodo hasta la hoja más baja.

Un nodo es visitado cuando acceso a los datos o realizar una operación sobre él.

Para “recorrer el árbol” debo haber visitado todos sus nodos.

Árbol desbalanceado es donde hay muchos más niveles que del otro.

actividad:”Jugamos Ninja”.

Recorrido de los árboles

Nodos: Representados con un círculo.

Aristas:Donde se conectan.

**Arboles Binarios:**

-Cada nodo tiene máximo 2 hijos.

9/10/19

**Árboles Binarios de Búsqueda:**

-No pueden haber nodos con el mismo valor.

-SI el hijo izquierdo es Menor que el nodo padre y su hijo derecho es Mayor que el nodo padre.

-Se pueden implementar con arreglos,listas enlazadas o “con estructuras de datos enlazadas”.

Operaciones en el árbol Binario de Búsqueda:

Contains(t dato), hace llamadas recursivas.

**Find Min/Max.**

**Insertar:** Hace un recorrido recursivo para llegar a donde se asigna el nodo.

**Delete:** Una vez se encuentra el nodo se consideran 3 posibles escenarios que el nodo sea (hoja,1 hijo o 2 hijos).

caso 1: Se elimina el nodo.

caso 2: El nodo hijo del nodo a eliminar pasa a ser hijo del nodo padre del nodo a eliminar:

caso 3: se reemplaza el nodo que se está eliminando por el nodo más pequeño del subárbol derecho o el mayor del que está a la izquierda.

actividad:Le chocamos la mano a Kenet.

11/10/19

Arboles AVL:

La altura de un árbol es igual a el largo del árbol-1.

Factor de balanceo= altura del subárbol derecho - subárbol izquierdo, los valores deben estar entre (-1,0,1).

Insertar: si se incumple la condición de balanceo se necesita realizar una rotación.

se verifica sólo los nodos que se encuentran en la ruta de la inserción del nodo.

Caso: Hijo derecho sub derecho.

Hijo izquierdo sub izquierdo.

Hijo derecho sub derecho.

Hijo izquierdo sub izquierdo.

Tipo de rotaciones: LL,RR o LR RL./ también existen las dobles rotaciones.

16/10/19

Splay Tree:

Implementación más eficiente para reducir el tiempo de acceso: cada vez que se visita un nodo el nodo pasa a ser la raíz (no hay altura ni balanceo).

Isert: El nodo insertado se convierte en la raíz.

Find: El nodo encontrado se vuelve raíz y el primero nodo en comparar en caso de no encontrar el nodo.Find(Max,Min)

Delete Min y Max:Ejecuta el Find y luego de eliminarlo su hijo pasa a ser raíz.

Remove: Utiliza el Find luego lo remueve y asigna como hijo a un sub arbol nodo del otro subárbol utilizando un Find (Max o Min).

Tipos de rotaciones: Zig, Zag ,Zig-Zag, Zag-Zig , Zig-Zig, Zag-Zag.

Heap Tree:

**Características:**

Es para implementar con cola de prioridad:

Rapida insercion O(logN).

**Requisitos:**

-Debe implementarse con arreglos.

-Es completo.(de arriba a abajo y de izquierda a derecha debe estar vacío después del primer vacío).

-Satisfaga la condición del heap.(todos los nodos hijos son menores que el nodo padre).

Pop: Remueve la raíz y se pone como raìz el último nodo, luego se hacen swaps para mandarlo abajo.

Insert: La primera posición disponible del arreglo.

N-ary Trees:

cantidad n de hijos.

Find: se accede al hijo mas izquierdo del nodo y se busca hacia el nodo izquierdo.

18/10/2019.

B-trees:

Se almacenan en el disco duro (gana almacenamiento pero pierde velocidad).

arbol de orden m.

árbol em-iario (Orden del árbol).

Todas las hojas deben estar en el mismo nivel.

Manejamos páginas en vez de nodos.

todas las páginas internas tienen un máximo de m ramificaciones y un mínimo de m medio ramificaciones.

La cantidad de llaves que hay a lo interno de cada página debe coincidir con la cantidad de ramificaciones -1.

Las ramificaciones deben seguir la lógica de los árboles binarios de búsqueda.

Un árbol B siempre está perfectamente balanceado.

Reduce la profundidad.

Insertar:

Divide la Llave en 2 y realiza la operación split para poner la llave en el lugar donde corresponde.

23/10/2019.

Árboles (Binarios) de expresión:

-Los nodos se llaman tokens.

-Las hojas son operandos, la raíz y los nodos internos son operadores.

-Cada subárbol es una sub expresión donde su raíz es el operador.

-Normalmente usados en compiladores para representar una expresión de un lenguaje de Programación en memoria.

-Se lee de izquierda a derecha.

-Formas de recorrer el árbol (infix,prefix,posfix).

Armar expresión a partir de arbol

Armar arbol a partir de una exprecion.

Stack: Operandores.

Cola de salida: Operandos.