

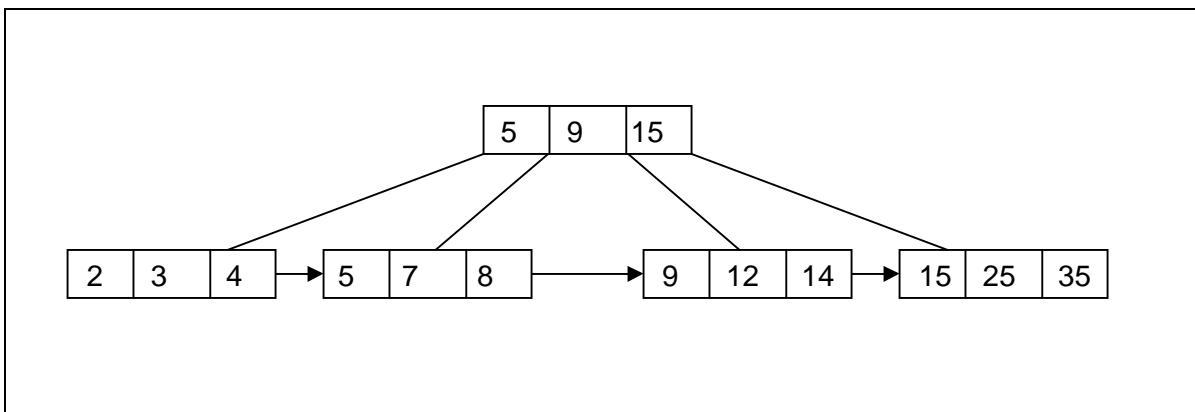
Laboratorio de Estructuras de Datos II

Segundo examen parcial

Fecha límite de entrega: Jueves 11 de junio de 2009, 3:59:59 p.m.

Los árboles B+ son estructuras de datos jerárquicas que se utilizan para almacenar y manipular datos ordenados de forma muy eficiente, ya que por su estructura y sus propiedades las inserciones y las eliminaciones se realizan en un tiempo logarítmico amortizado. Por esta razón, se utilizan para crear índices dentro de las bases de datos relacionales, para agilizar la búsqueda de registros.

Un nodo de un árbol B+ de orden n , al igual que en un árbol B, puede tener (n) hijos, por lo cual contendrá $n-1$ datos. Por ejemplo, un árbol B+ de orden 4 tiene la siguiente estructura:



En esta estructura (para el ejemplo un árbol de enteros, de orden 4), se debe garantizar que:

- Los datos dentro de cada nodo se encuentran ordenados de forma ascendente.
- Todos los datos de los hijos almacenados a la izquierda de cualquier dato deben ser menores que él y todos los datos de los hijos almacenados a la derecha de cualquier dato deben ser mayores o iguales que él.

Igualmente se deben cumplir las siguientes propiedades:

- Todas las hojas del árbol se encuentran en el mismo nivel
- Los nodos de un mismo nivel se encuentran unidos en una lista enlazada simple

Inserción en un árbol B+

El proceso de insertar un valor en un árbol B+ se realiza **siempre en los nodos hoja**, de la siguiente forma:

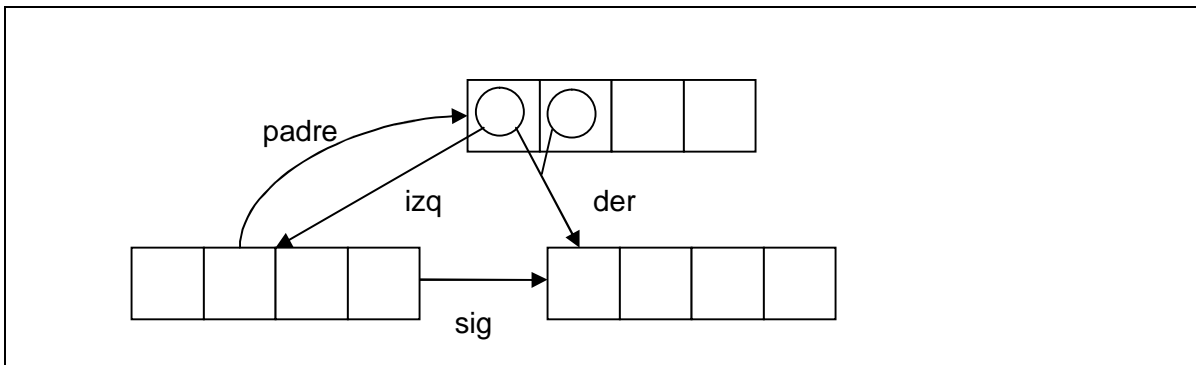
- Si el árbol está vacío, se debe crear un nuevo nodo B+ e insertar el nuevo valor. Este nodo se convierte en la raíz del árbol.
- Si el árbol no está vacío, primero se deberá verificar que el nuevo dato no exista dentro del árbol. Para ello se deberá realizar una búsqueda empezando desde la

- raíz. Si el nuevo dato no existe, el proceso de búsqueda deberá hallar el nodo hoja (B+) en el cual se va a insertar el nuevo nodo.
- Una vez encontrado el nodo hoja (B+) en el cual se almacenará el nuevo dato se pueden presentar dos casos:
 - La hoja (B+) tiene espacio para almacenar el nuevo dato. Se debe insertar el dato dentro del nodo B+ y terminar.
 - La hoja (B+) no tiene espacio: En este caso la hoja se deberá dividir en dos nodos B+. Se deberán disponer los valores (que estaban almacenados en la hoja más el nuevo valor) de forma ascendente y se tomará el valor de la mitad. Este valor se deberá llevar al nodo B+ padre (si no existe, se deberá crear uno). El hijo izquierdo del dato que se llevó al padre deberá contener los valores menores que él en la hoja original, y el hijo derecho deberá contener *el dato que se llevó al padre* y los datos mayores que él en la hoja original.
 - Si al llevar un dato al nodo padre no existe espacio para almacenarlo, se deberá dividir el nodo en dos (como en el paso descrito anteriormente), pero *el dato que se lleva al nodo padre no deberá ser almacenado en el hijo derecho*. En otras palabras, la copia del dato sólo se guarda en los nodos hoja.

Implementación de un nodo de árbol B+

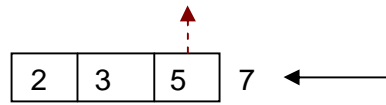
El nodo B+ se deberá implementar como una lista de nodos binarios, cada uno de los cuales tiene un apuntador izquierdo y derecho a sus nodos B+ hijos. Tenga en cuenta que los nodos binarios dentro del nodo B+ comparten el apuntador derecho e izquierdo con sus vecinos. También recuerde que todos los nodos B+ de un mismo nivel están conectados con el nodo B+ siguiente.

La siguiente gráfica ilustra cómo se debe implementar la estructura de datos para el nodo B+ (y el nodo binario dentro del nodo B+).



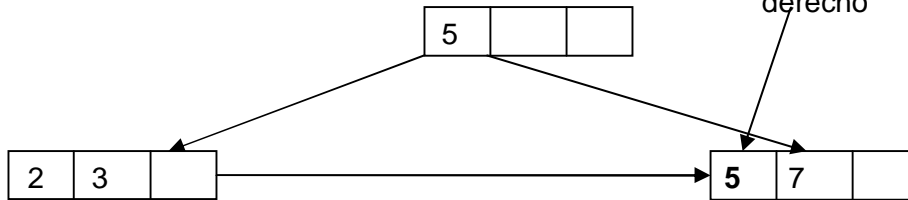
A continuación se ilustra gráficamente el proceso de insertar datos dentro de un árbol B+. Para el ejemplo, el árbol B+ es de orden 4 y almacena números enteros.

Insertar 2, 3, 5, 7

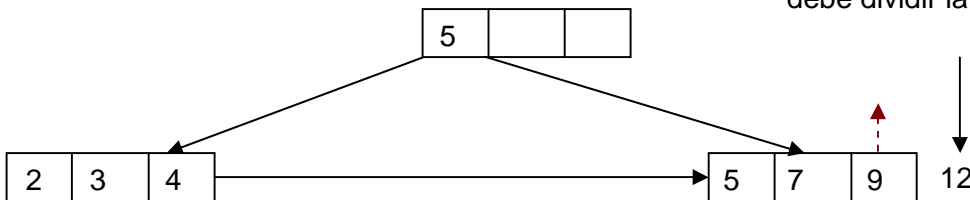


No hay espacio, se debe dividir la hoja.

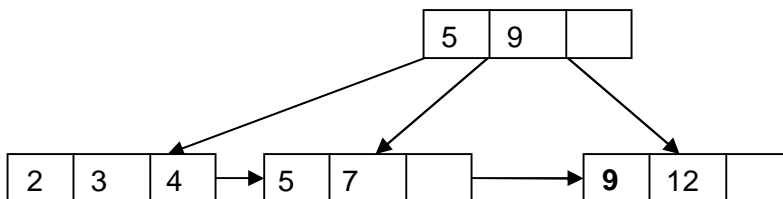
Cuando se divide una hoja, se conserva el dato que se subió al padre en su hijo derecho



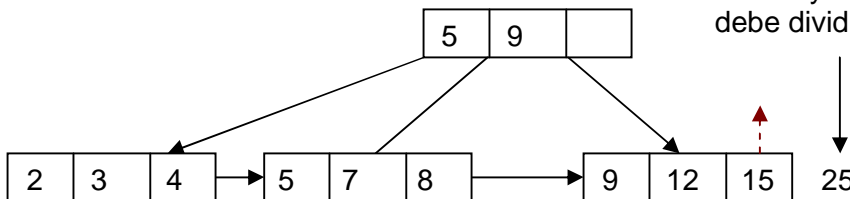
Insertar 4, 9, 12



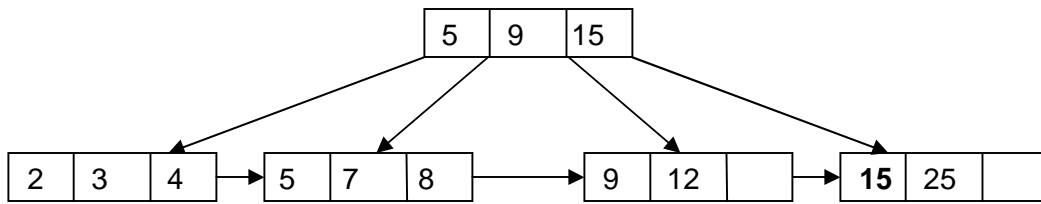
No hay espacio, se debe dividir la hoja.



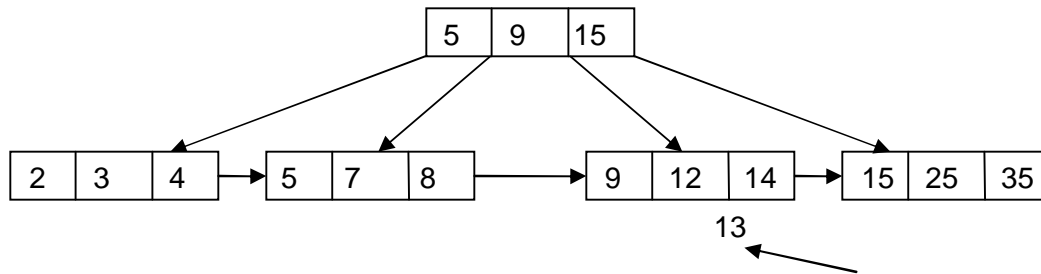
Insertar 8, 15, 25



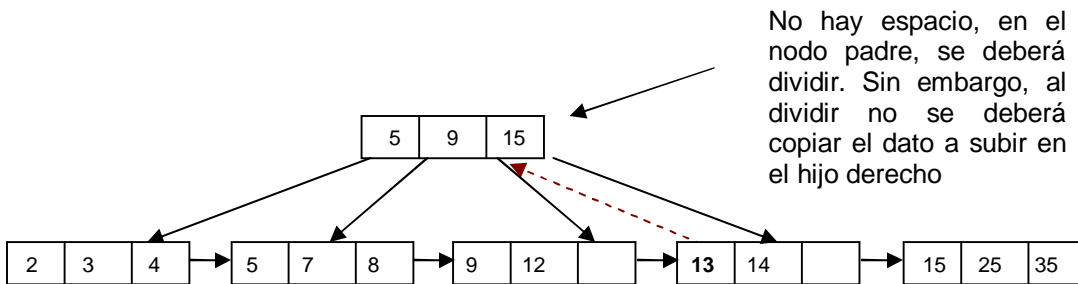
No hay espacio, se debe dividir la hoja.



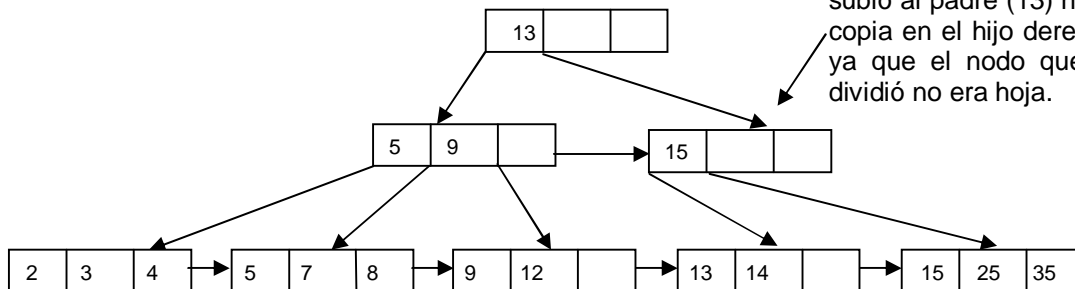
Insertar 14, 35, 13



No hay espacio, se debe dividir la hoja.



No hay espacio, en el nodo padre, se deberá dividir. Sin embargo, al dividir no se deberá copiar el dato a subir en el hijo derecho



Observe que en este caso el dato que se subió al padre (13) no se copia en el hijo derecho, ya que el nodo que se dividió no era hoja.

Problema

Se deberá implementar un programa que lea comandos desde la entrada estándar para insertar y buscar personas (cada una con cédula, primer nombre, segundo nombre, primer apellido y segundo apellido) en un árbol B+ que se encuentra inicialmente vacío. El programa deberá terminar cuando encuentre una línea en el archivo con la palabra “**salir**”. *El criterio de ordenamiento de las personas en el árbol B es su cédula.*

Cada vez que se busque una persona dentro del árbol se deberá imprimir el número de pasos que se realizaron para encontrarla, comparados con el número de pasos necesarios para buscar en una lista simple enlazada ordenada.

Los comandos que se deben implementar son:

orden <i>n</i>	Define el orden del árbol B+: El máximo número de hijos que puede tener cualquier nodo dentro del árbol.
cargar <i>archivo</i>	Cargar los datos del árbol B+ a partir de un archivo llamado <i>archivo</i>
buscar <i>#cedula</i>	Busca la persona con cuya cédula es <i>#cedula</i> dentro del árbol B+ e imprime el número de pasos que tuvo que realizar para encontrar el valor. En caso que no se encuentre la persona en el árbol también deberá imprimir el número de pasos que realizó. El número de pasos debe incluir la búsqueda dentro de cada nodo B+. Para cada caso se deberá imprimir también el número de pasos que se tuvo que realizar para encontrar el valor en una lista simple enlazada ordenada.
salir	Termina la ejecución.

El formato del archivo que se debe leer con el comando **cargar** es el siguiente:

```
#El archivo puede contener comentarios o líneas en blanco que deben ser ignorados
#Cada línea describe a una persona. Los campos para cada persona están separados
#por espacios, y son los siguientes:
#cédula primer_nombre segundo_nombre primer_apellido segundo_apellido
#Los únicos campos obligatorios para cada persona son el número de la cédula, el
#primer nombre y el primer apellido. Los campos faltantes se representan con “.”
```

```
13500710 juan . diaz .
10200340 pedro pablo perez .
```

Salida del programa

El programa sólo deberá imprimir cuando reciba un comando “buscar”. Por ejemplo, para el comando:

```
buscar 13500710
```

El programa deberá imprimir si la persona existe o no dentro del árbol, el número de comparaciones que se realizaron en el árbol B+, y el número de comparaciones que se tendrían que realizar en una lista enlazada simple ordenada.