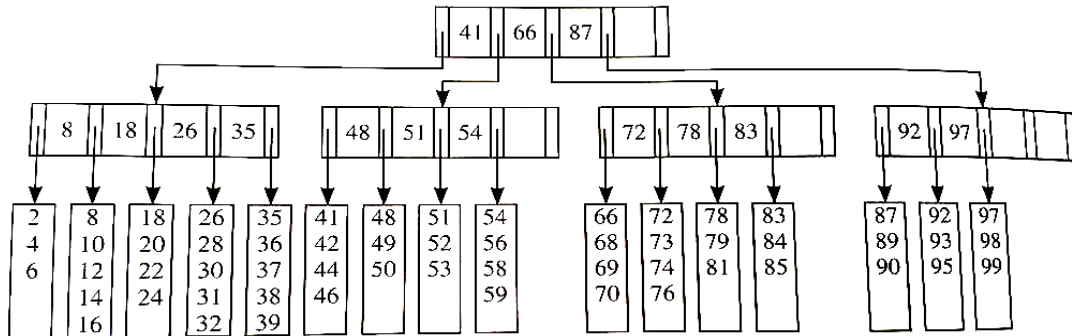


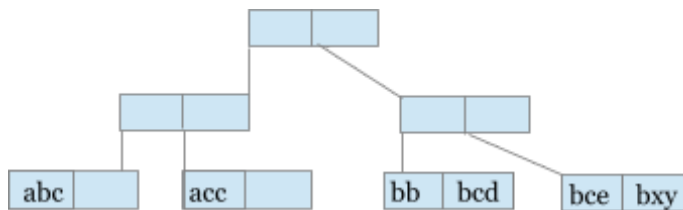
Algoritmos y Estructuras de Datos III
Tarea 6 - Unidad 4 - **Ordenación Externa**

Parte I - Ejercicios teóricos (10pts) (Alzar en archivo PDF)

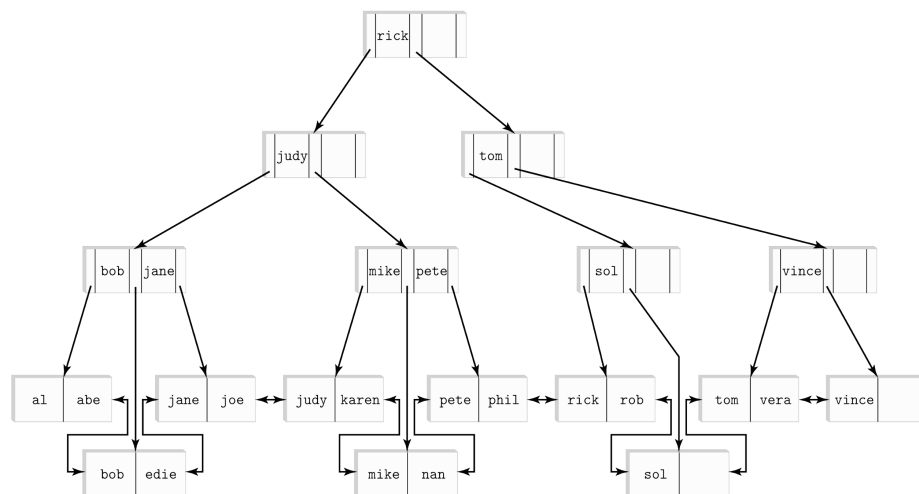
- Se desea construir un B+Arbol para registros en donde la clave ocupa 64 bytes y la dirección a nodos del árbol ocupa 4 bytes. Calcular los valores M y L ideales para 2 discos donde el tamaño de bloque es 4096 y 8192 respectivamente. Considere que existen datos aparte de la clave que ocupa 96 bytes. (2p)
- Considere el ejemplo presentado en el libro de Weiss sobre B+Arbol (Árbol B de orden M). Con ese ejemplo y considerando las observaciones del autor realice los siguientes ejercicios: (4p)



- Agregue el valor 100 al árbol al árbol de la figura
 - Busque una secuencia de inserciones que haga crecer al árbol en un nivel más.
 - Elimine el valor 6, luego el valor 50 y luego el valor 81 de la figura
 - Indique con un sencillo pseudocódigo como haría para imprimir todos los datos en forma ordenada de un B+Arbol.
- Considere el B-Arbol parcialmente indicado en la siguiente figura: (2p)



- Complete los nodos internos sin agregar nuevas claves.
 - Agregue la clave **bbb**. Muestre el árbol cambiado.
- Dibuje el B+Arbol que resulta de insertar alice, betty, carol, debbie, edith y zelda en el presentado a continuación (2p) Para el desarrollo del ejercicio, obvie los enlaces existentes que conectan a las hojas entre si.



Rúbrica	Puntaje
Item 1 correcto	2
Item 2 correcto	4
Item 3 correcto	2
Item 4 correcto	2

Parte II - Ejercicios de programación

Se debe entregar el código fuente de cada ejercicio en la clase que implementa y la clase main para las pruebas, cada uno en una tarea VPL separada. Cantidad máxima de archivos por cada ejercicio: 2.

Ejercicio 1 (30 pts)

Descargue el código de B-árbol disponible en:

<https://algs4.cs.princeton.edu/code/edu/princeton/cs/algs4/BTree.java.html>

Luego realice las siguientes operaciones:

- Modifique los métodos que considere necesarios para que el código implemente un B+Árbol.
- Implemente el método **rangePrint(value1,value2)** donde (**value1 < value2**) que imprime los valores en los rangos dados si los mismos se encuentran en el B+Árbol. Si los rangos no son válidos o no existen se debe lanzar una excepción.

Rúbrica	Puntaje
Implementación correcta, sin errores de compilación y de acuerdo a restricciones de las operaciones solicitadas de la implementación de B+Árbol	15
Implementación correcta, sin errores de compilación y de acuerdo a restricciones de las operaciones solicitadas de la implementación de rangePrint	10
Provisión de datos de prueba	3
Comentarios aclaratorios	2

Ejercicio 2 (10 pts)

Descargue el código de B-Árbol disponible en:

<https://algs4.cs.princeton.edu/code/edu/princeton/cs/algs4/BTree.java.html>

Luego modifique el valor de M de manera a que pueda ingresarse como parámetro del ejecutable, por ejemplo:

```
c:/> java -jar barbol.jar 10
```

Debe poder generar un B-Árbol con M=10. Como restricción, solamente debe poder ingresarse valores de M pares y mayores a 2, en caso contrario se debe lanzar una excepción.

Rúbrica	Puntaje
Implementación correcta, sin errores de compilación y de acuerdo a restricciones de las operaciones solicitadas de la implementación de B-Árbol	6
Provisión de datos de prueba	2
Comentarios aclaratorios	2