Cấu trúc dữ liệu và Thuật toán

Tìm kiếm

Cây Đỏ- Đen và Các cây cân bằng động khác

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

Tóm tắt

- Cây Đỏ - Đen (Red-Black)
  - Một thuật toán phức tạp
  - Cho $O(\log n)$ với Thêm(addition), Xóa(deletion) và Tìm(search)
- Các kỹ thuật phần mềm
  - Biết về thuật toán
  - Biết khi nào có thể dùng chúng
  - Biết về hiệu suất
  - Biết nơi có thể áp dụng
  - Đủ thông minh để dùng cho các máy phát minh...
    - Họ sử dụng lại bộ mã
    - Vì thế họ cần nhiều thời gian hơn cho
      - Chuyến đi, ăn, uống, ...

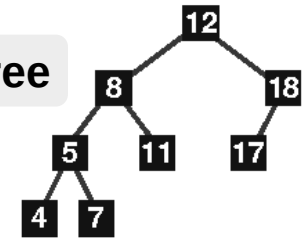
      ☺Bất cứ những gì khác mà tôi sẽ cho vào đây.

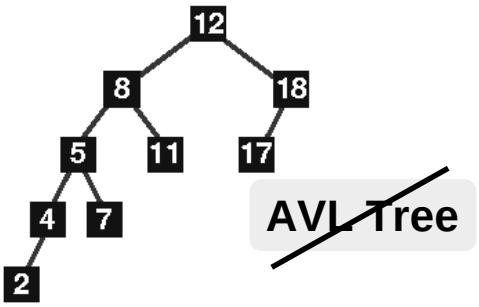
This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

Cây AVL và các cây cân bằng khác

- **Cây AVL**
  - **Thuật giải cây cân bằng đầu tiên**
  - **Khám phá bởi: Adelson-Velskii và Landis**
- **Tính chất**
  - **Cây nhị phân tìm kiếm**
  - **Độ cao con trái và con phải chênh lệch nhiều nhất là 1**
  - **Các cây con cũng là cây AVL**

AVL Tree

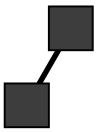


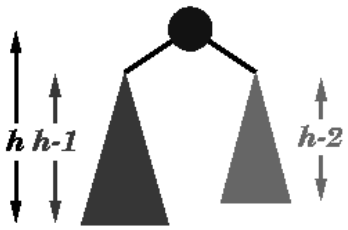


~~AVL Tree~~

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

Cây AVL – Chiều cao

- **Định lý**
  - Một cây AVL có chiều cao h có *ít nhất* $F_{h+3}+1$ node
- **Chứng minh**
  - Để S_h là kích cỡ của cây AVL *nhỏ nhất* với chiều cao h
  - Rõ ràng, $S_0 = 1$  and $S_1 = 2$ 
  - Hơn nữa, $S_h = S_{h-1} + S_{h-2} + 1$
  - Một cây có chiều cao cực tiểu phải bao gộp nhiều cây có chiều cao nhỏ với độ lệch tối đa là 1
  - Do đó ..
    - $S_h = F_{h+3}+1$



This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

Cây AVL – Chiều cao

- Bây giờ, tổng quát cho i, $F_{i+1} / F_i = \phi$,

  trong đó $\phi = \frac{1}{2}(1 + \sqrt{5})$

  hoặc $F_i \approx c\,(\frac{1}{2}(1 + \sqrt{5}))^i$

  - $S_h = F_{h+3} + 1 = O(b^h)$
- $n \geq S_h$, vì thế n là $\Omega(b^h)$

  và $h \leq \log_b n$ *hoặc* h is $O(\log n)$

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

Cây AVL – Chiều cao

- Bây giờ, tổng quát cho i, $F_{i+1} / F_i = \phi$,

  trong đó $\phi = \frac{1}{2}(1 + \sqrt{5})$

  hoặc $F_i \approx c\,(\frac{1}{2}(1 + \sqrt{5}))^i$

  - $S_h = F_{h+3} + 1 = O(b^h)$
- $n \geq S_h$, vì thế n là $\Omega(b^h)$

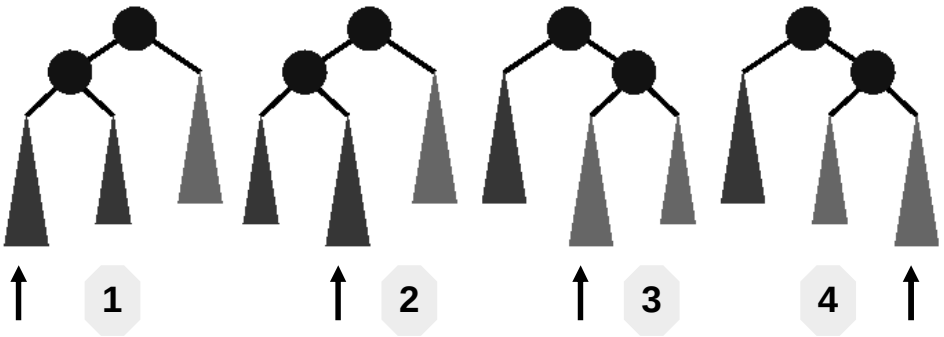
  và $h \leq \log_b n$ hoặc h là $O(\log n)$
- Trong trường hợp này, chỉ ra
  - $h \leq \lceil 1.44 \log_b (n+2) - 1.328 \rceil$

h không tồi hơn 44% cao hơn so với tối ưu

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

Cây AVL – Tái cân bằng

- **Chèn các lá vào cây non-AVL**
  - **4 trường hợp**

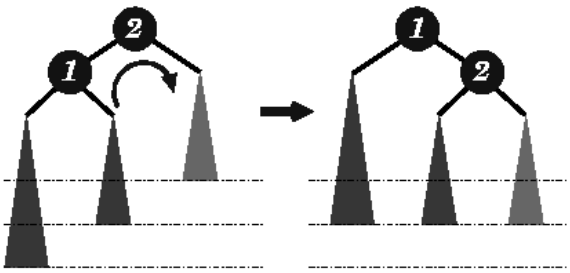


  - **1 và 4 là các ảnh đối xứng**
  - **2 và 3 là các ảnh đối xứng**

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

Cây AVL – Tái cân bằng

- **Trường hợp 1 được giải bởi phép quay**

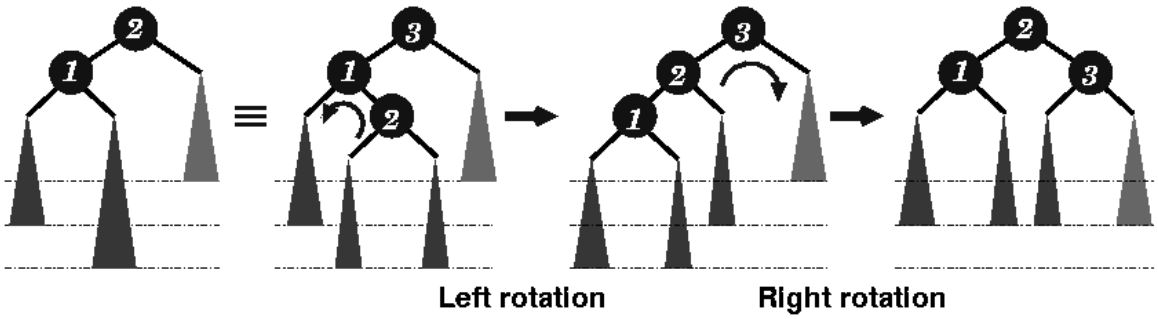


- **Trường hợp 4 là quay một ảnh đối xứng**

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

Cây AVL – Tái cân bằng

- **Trường hợp 2 cần một phép quay kép (*double*)**



  - **Trương hợp 3 là phép quay ảnh đối xứng**

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

Cây AVL – Cấu trúc dữ liệu

- Các cây AVL có thể được thực hiện với một cờ để báo hiệu trạng thái cân bằng

```
typedef enum { LeftHeavy, Balanced, RightHeavy }
                            BalanceFactor;
struct AVL_node {
    BalanceFactor bf;
    void *item;
    struct AVL_node *left, *right;
    }
```

- Chèn (Insertion)
  - Chèn một node mới (giống bất cứ cây nhị phân nào)
  - Việc cân bằng lại cây (duyệt ngược lên) cần thiết phục hồi các tính chất của AVL

This document was created using SOLID CONVERTER PDF To remove this message, purchase the product at www.SolidDocuments.com

Các cây động - Red-Black hoặc AVL

- **Chèn (Insertion)**
  - **AVL : hai bước thực hiện trên cây**
    - **Duyệt xuống để chèn thêm node**
    - **Duyệt lên để tái cân bằng (re-balance)**
  - **Red-Black : hai bước thực hiện trên cây**
    - **Duyệt xuống để chèn thêm node**
    - **Duyệt lên để tái cân bằng (re-balance)**

nhưng Red-Black phổ biến hơn??

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

Các cây động – Cảnh báo

- **Chèn (Insertion)**
  - **Nếu bạn đọc Cormen *et al,***
    - **Không có một lý do nào để thích một cây đỏ- đen**
  - ***Tuy nhiên,* trong sách của Weiss**

    M A Weiss, *Algorithms, Data Structures and Problem Solving with C++*, Addison-Wesley, 1996
  - **Bạn tìm thấy rằng bạn có thể cân bằng cây red-black**
    Tổng kết lại !
  - **Tạo một cây red-black nhiều hiệu quả hơn AVL**
    Nếu mã code là hợp lý!!!

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

Các cây động – Cảnh báo

- **Chèn (Insertion)**
  - Nếu bạn đọc Cormen *et al,*
    - Không có một lý do nào để thích một cây đỏ- đen
  - *Tuy nhiên,* trong sách của Weiss

    M A Weiss, *Algorithms, Data Structures and Problem Solving with C++*, Addison-Wesley, 1996
  - Bạn tìm thấy rằng bạn có thể cân bằng cây red-black
    Tổng kết lại !
  - Tạo một cây red-black nhiều hiệu quả hơn AVL
    Nếu mã code là hợp lý!!!

Yêu cầu: Bạn cần thiết đọc các tài liệu!

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

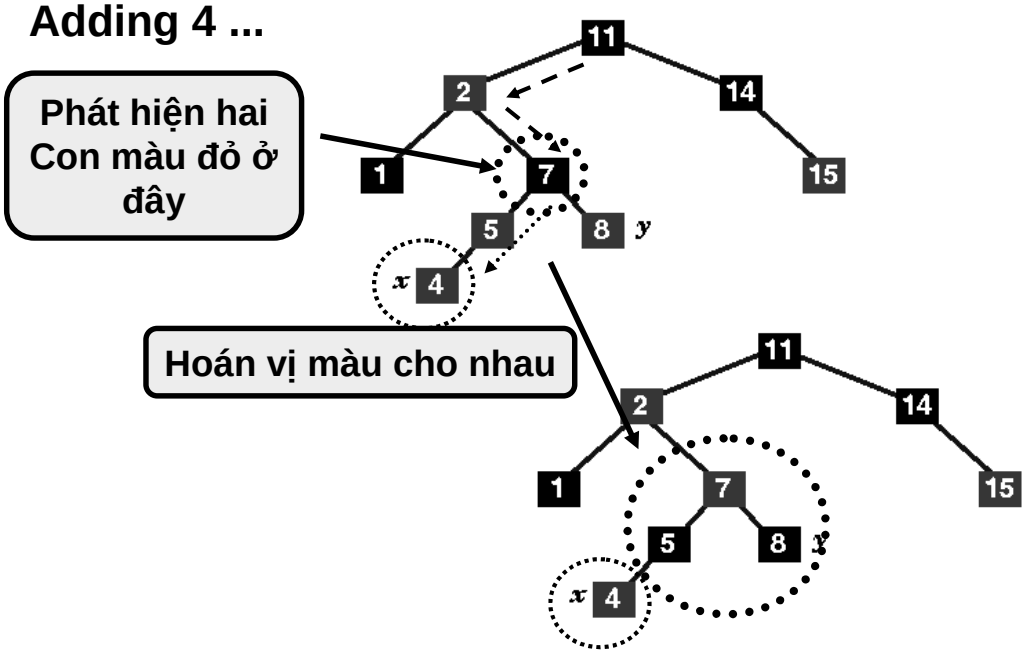
Các cây động – Cảnh báo

- **Tổng kết cho "chèn" Insertion**
  - **Khi bạn đi đến phía dưới của một cây,**
    nếu bạn tìm thấy một node với hai con red,
    tô nó màu red và con của nó màu black
  - **Điều này không biến đổi số lượng các node đen trong bất cứ nhánh nào**
  - **Nếu cha của node là mầu red,**
    một phép quay là cần thiết ...
  - **Có thể cần một phép quay đơn hoặc quay kép**

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

Cây – Chèn (Insertion)

- **Adding 4 ...**

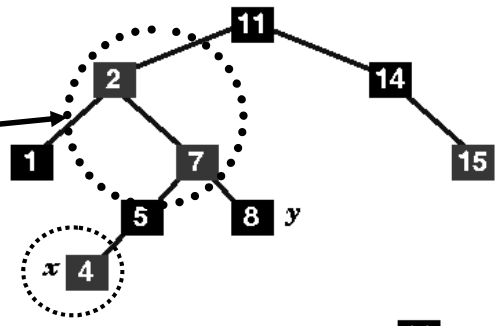


This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

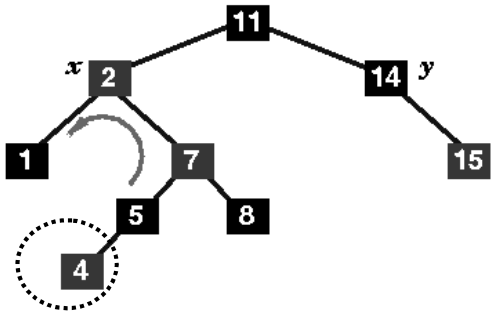
Cây – Chèn (Insertion)

- **Adding 4 ...**

**Một dãy node đỏ,
Vi phạm
Tính chất red-black**



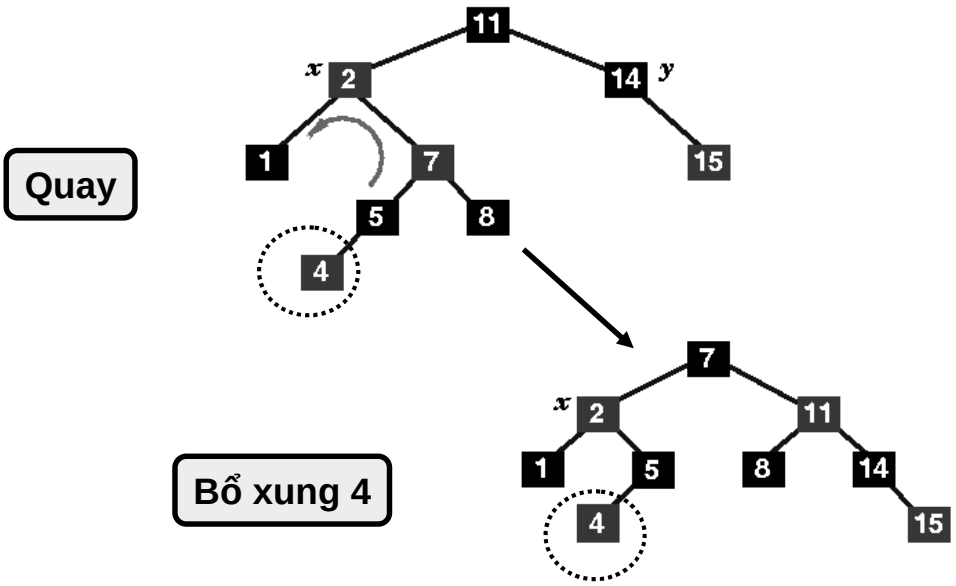
Quay



This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

Cây – Chèn (Insertion)

- **Adding 4 ...**



This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

Cây cân bằng – Chưa có nhiều biến đổi

- **Cơ bản có cùng các ý tưởng**
  - **2-3 Trees**
  - **2-3-4 Trees**
    - **Trường hợp dặc biệt của cây m-way!**
    - **Số lượng các con cho một node biến thiên**

    ⇐Thủ tục thực hiện phức tạp hơn
    - **2-3-4 trees**
      - **Chiếu về cây red-black**

      ∴ Có lẽ hữu dụng khi hiểu một cây red-black

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

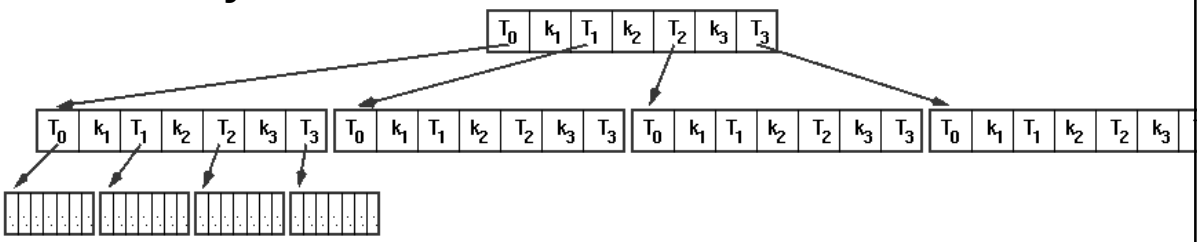
Tổng kết

- **Cây AVL**
  - **Đầu tiên là cây cân bằng động**
  - **Chiều cao thuộc 44% so với điều kiện thuận lợi nhất**
  - **Tái cân bằng với các góc quay**
  - **O(log n)**
    - **Hiệu quả thấp hơn so với cây red-black**
- **2-3, 2-3-4 trees**
  - **m-way trees – chưa có nhiều biến đổi**
  - **2-3-4 trees được chiếu về cây red-black**

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

Cây m nhánh (m-way trees)

- **Chỉ hai con cho một node?**
- **Rút gọn độ sâu của cây đến $O(\log_m n)$ với *m*-way trees**



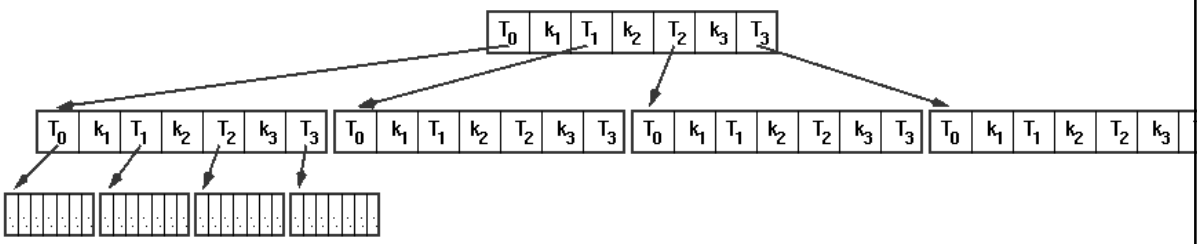
- ***m* con, *m*-1 khóa cho node**
- ***m* = 10 : 10^6 khóa trong 6 mức *khác* 20 cho cây nhị phân**
- ***nhưng* .....**

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

Cây m nhánh (m-way trees)

- **Nhưng bạn phải tìm kiếm *m* khóa trong mỗi node!**
- **Đánh đổi giữa số mức và thời gian tìm kiếm!**

➔ Chỉ là một sự hiếu kỳ?



This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

Cây B(Block) (B-trees)

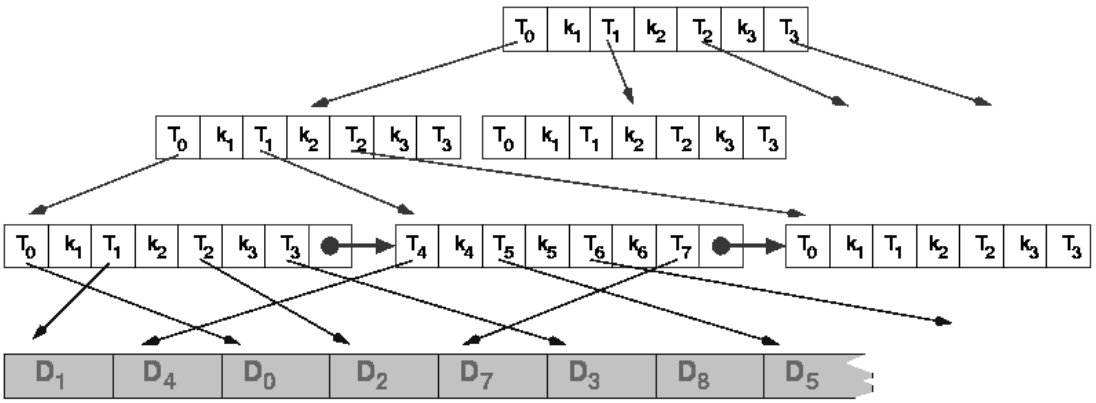
- Tất cả các lá nằm trên cùng một mức
- Tất cả các node ngoại trừ gốc và các lá có:
  - Ít nhất $m/2$ con
  - Nhiều nhất m con
- Cây B+ (B+ trees)
  - Tất cả các khóa trong các node là giả
  - Chỉ các khóa trong các lá nhận giá trị thực "real"
  - Liên kết các lá
    - Có khả năng duyệt hết danh sách *theo thứ tự giữa* không cần thông qua các node cao hơn.

Mỗi node chứa ít nhất một nửa số lượng khóa

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

Cây B+

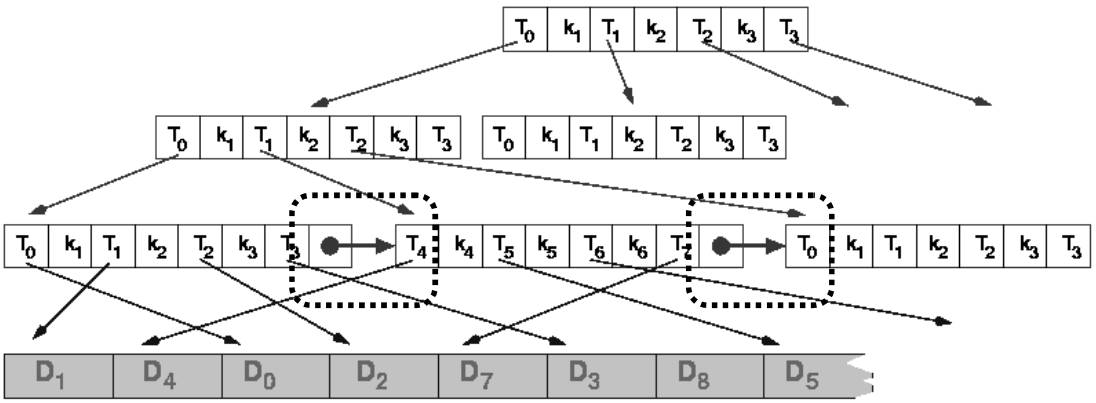
- **Cây B+**
  - **Tất cả các khóa trong các node là giả**
  - **Chỉ các khóa trong các lá nhận giá trị thực "real"**
  - **Các bản ghi dữ liệu được giữ trong các vùng riêng**



This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

Cây B+ - Duyệt theo thứ tự giữa

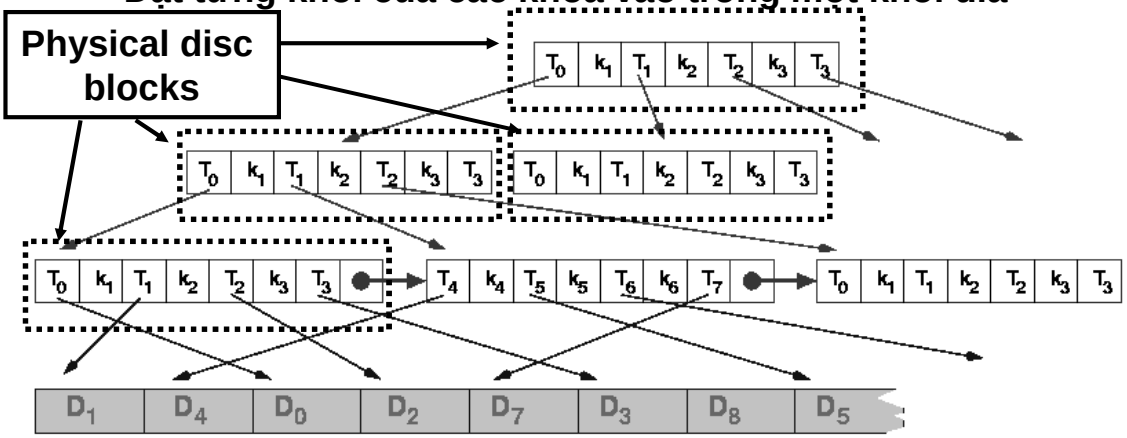
- **Cây B+ (B+ trees)**
  - **Liên kết các lá**
    - **Có khả năng duyệt hết danh sách *theo thứ tự giữa* không cần thông qua các node cao hơn.**



This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

Cây (B+) – Sử dụng

- **Sử dụng - Cơ sở dữ liệu lớn**
  - **Đọc một khối đĩa chậm hơn nhiều so với đọc bộ nhớ (~ms *vs* ~ns)**
  - **Đặt từng khối của các khóa vào trong một khối đĩa**



This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

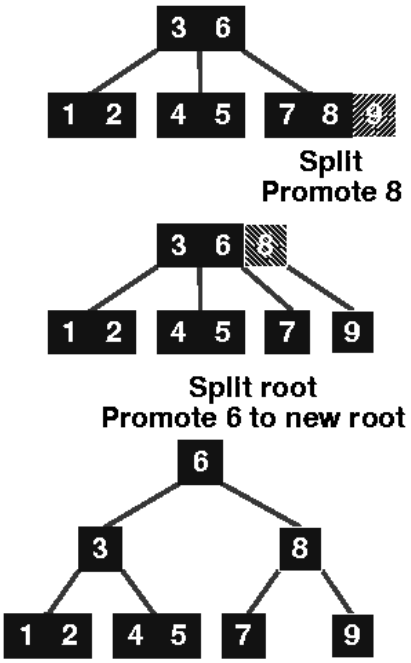
Cây B (B-trees) - Chèn

- **Chèn (Insertion)**
  - **Tính chất cây B (B-tree): một khối có ít nhất một nữa số khóa**
  - **Chèn vào trong khối với *m khóa***
    - **Tràn khối**
    - **Phân tách khối**
    - **Đẩy một khóa**
    - **Phân tách cha mẹ nếu cần thiết**
    - **Nếu gốc bị tách, cây được đặt ở mức sâu hơn**

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

Cây B – Chèn

- **Chèn**
  - **Chèn 9**
  - **Node bị tràn, phân tách nó**
  - **Đẩy node giữa (8)**
  - **Gốc bị tràn, phân tách nó**
  - **Đẩy node giữa (6)**
  - **Node gốc mới hình thành**
  - **Chiều cao tăng 1**



This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com

Cây B trên đĩa

- **Các khối đĩa**
  - **512 - 8k bytes**

  ∴ 100s of keys

  ←Dùng tìm kiếm nhị phân cho các khối
- **Tổng quát**
  - $O(\log n)$
  - **Làm hợp với phần cứng !**
- **Thủ tục xóa tương tự (Deletion)**
  - **Tuy nhiên, phải hội nhập các khối (block) để bảo đảm tính chất B-tree**
    (ít nhất bằng nửa số lượng khóa)

This document was created using SOLID CONVERTER PDF
To remove this message, purchase the product at www.SolidDocuments.com