

Cấu trúc dữ liệu và giải thuật

CẤU TRÚC CÂY

Giảng viên:

Văn Chí Nam – Nguyễn Thị Hồng Nhung – Đặng Nguyễn Đức Tiến

Nội dung trình bày

2

Khái niệm

Phép duyệt cây và Biểu diễn cây

Cây nhị phân và Cây nhị phân tìm kiếm

Cây AVL

Cây AA

Cấu trúc dữ liệu và giải thuật - HCMUS 2011

3

Khái niệm

Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Một số thuật ngữ

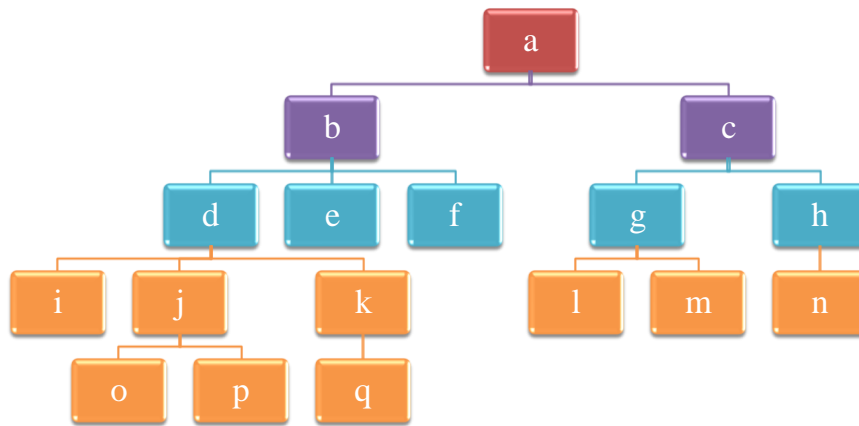
4

- Tree
- Search tree
- Binary search tree
- Balanced tree
- AVL tree
- AA tree
- Red-Black tree
- ...

Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Cây tổng quát

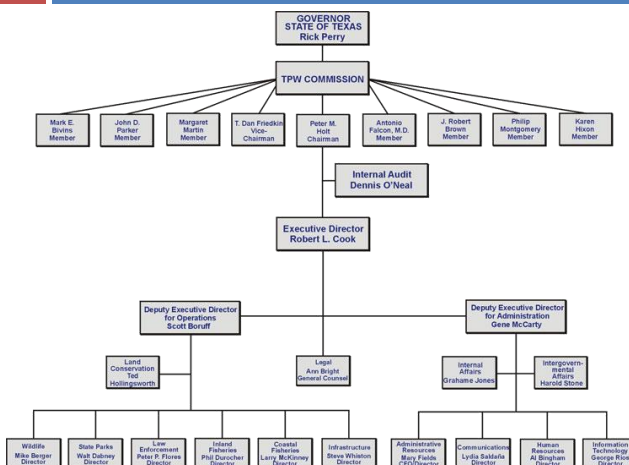
5



Cấu trúc dữ liệu và giải thuật - HCMUS 2011

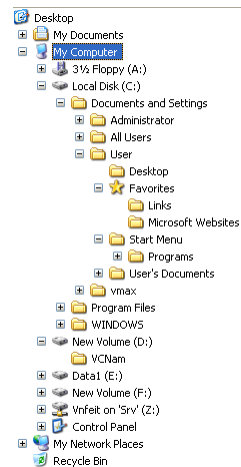
Cây tổng quát

6



Sơ đồ tổ chức

Cấu trúc dữ liệu và giải thuật - HCMUS 2011



Cây thư mục

Định nghĩa

7

Cây (cây có gốc) được xác định đệ quy như sau:

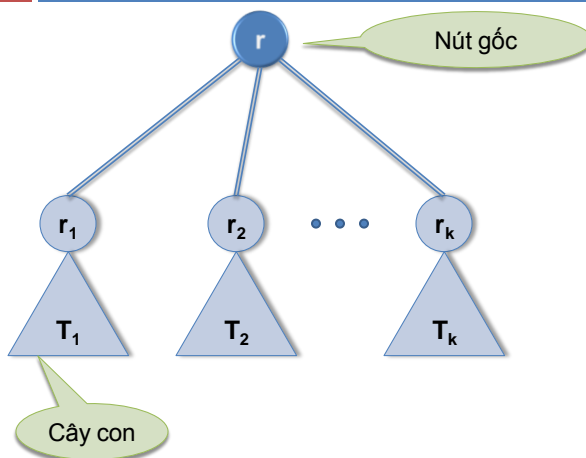
1. Tập hợp gồm 1 **đỉnh** là một cây. Cây này có **gốc** là đỉnh duy nhất của nó.
2. Gọi T_1, T_2, \dots, T_k ($k \geq 1$) là các cây không cắt nhau có gốc tương ứng r_1, r_2, \dots, r_k .

Giả sử r là một đỉnh mới không thuộc các cây T_i . Khi đó, tập hợp T gồm đỉnh r và các cây T_i tạo thành một cây mới với gốc r . Các cây T_1, T_2, \dots, T_k được gọi là cây con của gốc r .

Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Định nghĩa

8



Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Các khái niệm

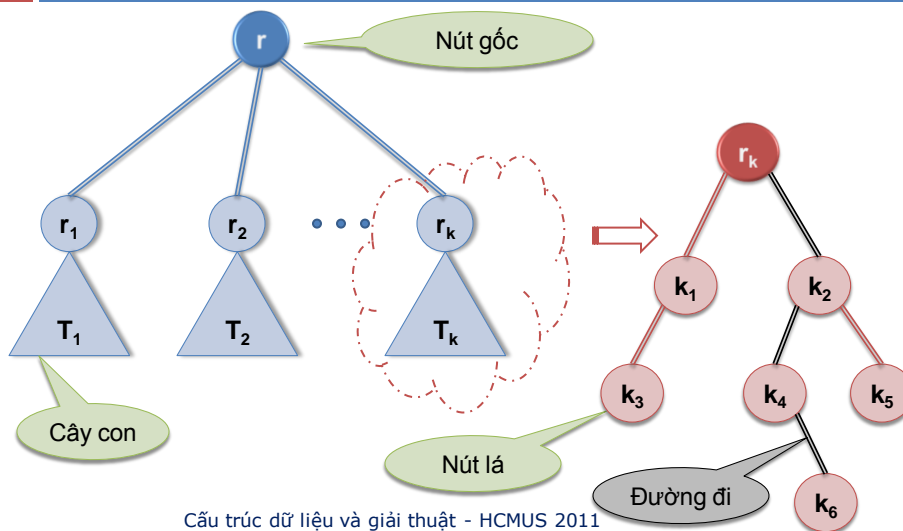
9

- node: đỉnh
- root: gốc cây
- leaf: lá
- inner node/internal node: đỉnh trong
- parent: đỉnh cha
- child: đỉnh con
- path: đường đi

Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Các khái niệm

10



Các khái niệm

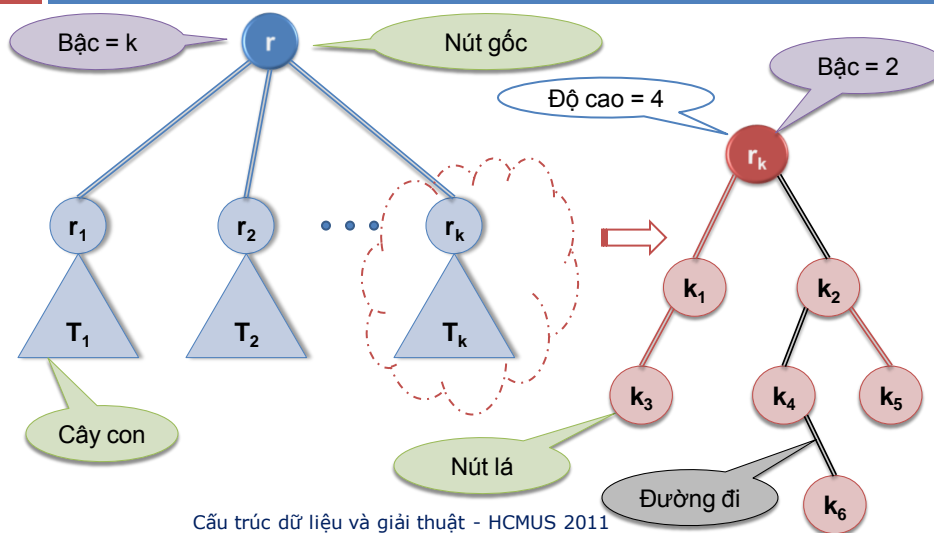
11

- **degree/order: bậc**
 - ▣ Bậc của node: Số con của node
 - ▣ Bậc của cây: bậc lớn nhất trong số các con
- **depth/level: độ sâu/mức**
 - ▣ Mức (độ sâu) của node: Chiều dài của đường đi từ node gốc đến node đó cộng thêm 1.
- **height: chiều cao**
 - ▣ Chiều cao cây:
 - Cây rỗng: 0
 - Cây khác rỗng: Mức lớn nhất giữa các node của cây

Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Các khái niệm

12



Cấu trúc dữ liệu và giải thuật - HCMUS 2011

13

Phép duyệt cây

Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Phép duyệt cây

14

- Đảm bảo đến mỗi node trên cây **chính xác một lần** một cách **có hệ thống**.
- Nhiều thao tác xử lý trên cây cần phải sử dụng đến phép duyệt cây.
- Các phép cơ bản:
 - ▣ Duyệt trước (Pre-order)
 - ▣ Duyệt giữa (In-order)
 - ▣ Duyệt sau (Post-order)

Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Phép duyệt cây

15

Tìm cha một đỉnh.

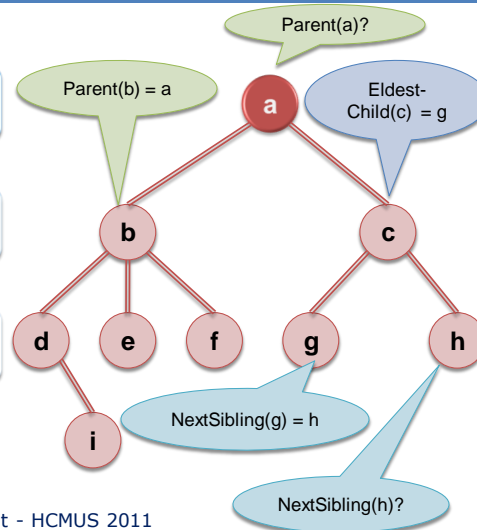
- $Parent(x)$

Tìm đỉnh con trái nhất.

- $EldestChild(x)$

Tìm đỉnh kề phải.

- $NextSibling(x)$



Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Phép duyệt cây

16

Duyệt trước

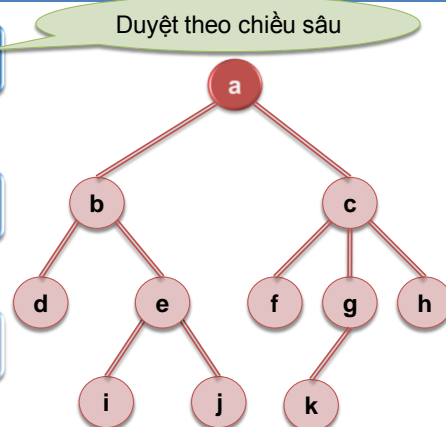
- $abdeijc f g k h$

Duyệt giữa

- $dbiejafckgh$

Duyệt sau

- $dijebfkghca$



Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Phép duyệt cây

17

Pre-order

```
void Preorder(NODE A)
{
    NODE B;
    Visit(A);
    B = EldestChild(A);
    while (B != Ø) {
        Preorder(B);
        B = NextSibling(B);
    }
}
```

Post-order

```
void Postorder(NODE A)
{
    NODE B;

    B = EldestChild(A);
    while (B != Ø) {
        Postorder(B);
        B = NextSibling(B);
    }
    Visit(A);
}
```

Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Phép duyệt cây

18

In-Order

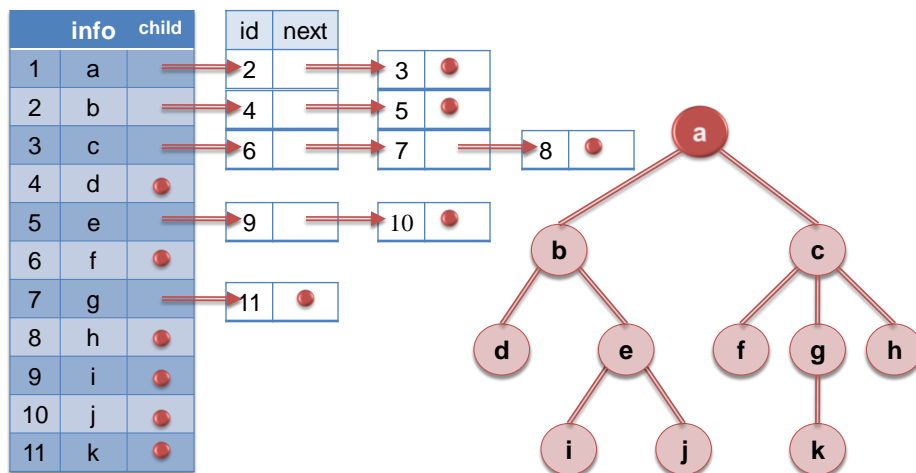
```
void Inorder(NODE A)
{
    NODE B;
    B = EldestChild(A);
    if (B != Ø) {
        Inorder(B);
        B = NextSibling(B);
    }
    Visit(A);
    while (B != Ø) {
        Inorder(B);
        B = NextSibling(B);
    }
}
```

Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Biểu diễn cây

Cấu trúc dữ liệu và giải thuật - HCMUS 2011

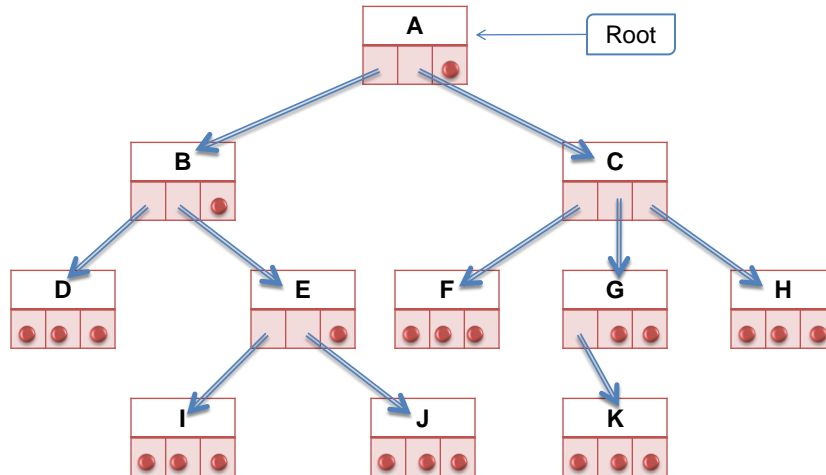
Bảng danh sách cây con



Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Bảng danh sách cây con

21

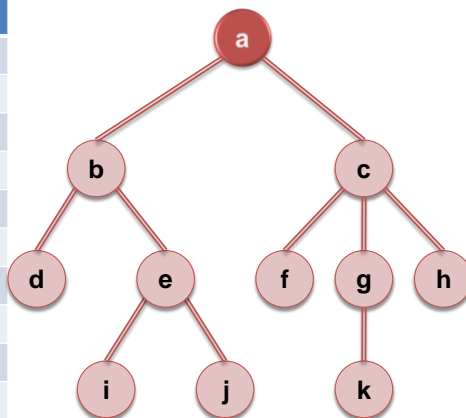


Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Bảng đỉnh trái nhất và đỉnh kề phải

22

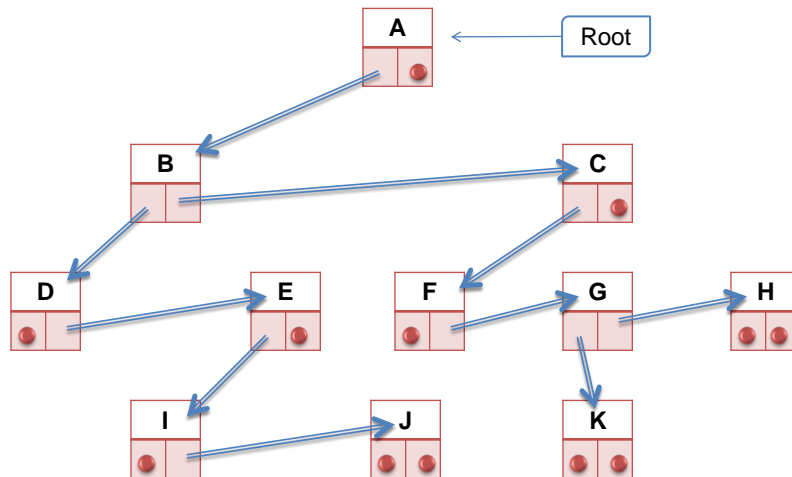
	Info	Eldest Child	Next Sibling
1	a	2	0
2	b	4	3
3	c	6	0
4	d	0	5
5	e	9	0
6	f	0	7
7	g	11	8
8	h	0	0
9	i	0	10
10	j	0	0
11	k	0	0



Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Bảng đỉnh trái nhất và đỉnh kề phải

23

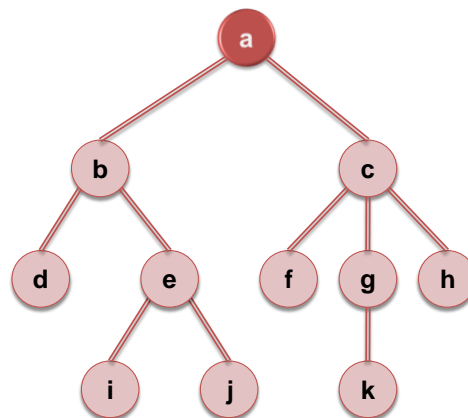


Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Bảng cha mỗi đỉnh

24

	Info	Parent
1	a	0
2	b	1
3	c	1
4	d	2
5	e	2
6	f	3
7	g	3
8	h	3
9	i	5
10	j	5
11	k	7



Cấu trúc dữ liệu và giải thuật - HCMUS 2011

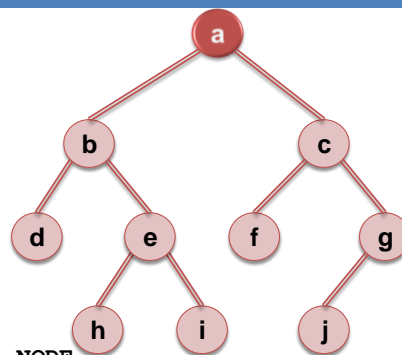
Cây nhị phân

Binary tree

Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Cây nhị phân

- Là cây mà mỗi đỉnh có bậc tối đa bằng 2.
- Các cây con được gọi là cây con trái và cây con phải.
- Có toàn bộ các thao tác cơ bản của cây.



```

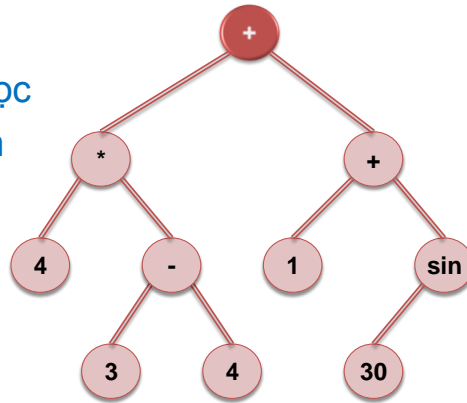
struct NODE
{
    Data key;
    NODE *pLeft;
    NODE *pRight;
};
  
```

Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Một số ứng dụng

27

- Cây tổ chức thi đấu
- Cây biểu thức số học
- Lưu trữ và tìm kiếm thông tin.



Cây biểu thức:
 $4 * (3 - 4) + (1 + \sin(30))$

Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Cây nhị phân tìm kiếm

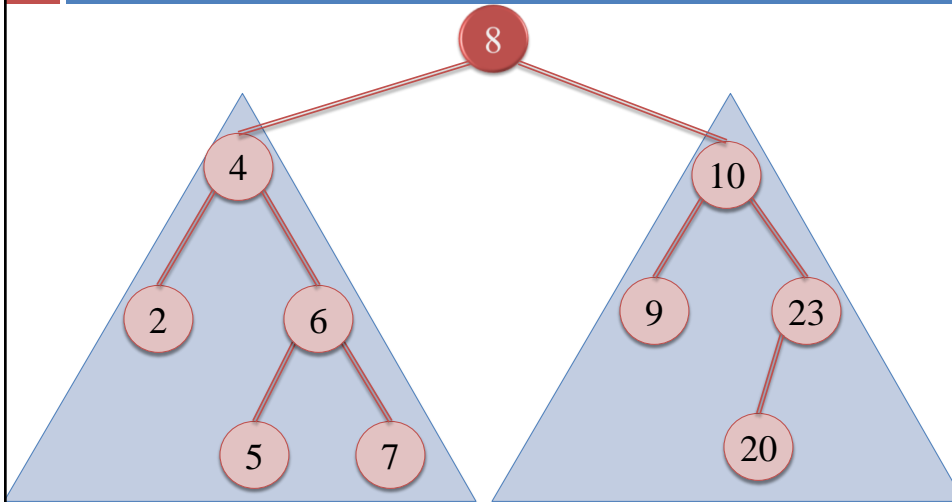
28

- Cây nhị phân tìm kiếm là cây nhị phân thỏa mãn các điều kiện sau:
 1. Khóa của các đỉnh thuộc cây con trái nhỏ hơn khóa gốc.
 2. Khóa của gốc nhỏ hơn khóa các đỉnh thuộc cây con phải.
 3. Cây con trái và cây con phải của gốc cũng là cây nhị phân tìm kiếm.

Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Cây nhị phân tìm kiếm

29



Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Cây nhị phân tìm kiếm

30

◉ Đặc điểm:

- ▣ Có thứ tự
- ▣ Không có phần tử trùng
- ▣ Dễ dàng tạo dữ liệu sắp xếp, và tìm kiếm

Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Thao tác trên cây nhị phân tìm kiếm

Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Các thao tác

32

- Thêm phần tử (khóa)
- Tìm kiếm phần tử (khóa)
- Xóa phần tử (khóa)
- Sắp xếp
- Duyệt cây
- Quay cây

Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Thêm phần tử

33

- Bước 1: Bắt đầu từ gốc
- Bước 2: So sánh dữ liệu (khóa) cần thêm với dữ liệu (khóa) của node hiện hành.
 - ▣ Nếu bằng nhau => Đã tồn tại. Kết thúc
 - ▣ Nếu nhỏ hơn => Đi qua nhánh trái, Tiếp bước 2.
 - ▣ Nếu lớn hơn => Đi qua nhánh phải, Tiếp bước 2.
- Bước 3: Không thể đi tiếp nữa => Tạo node mới với dữ liệu (khóa) cần thêm. Kết thúc

Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Tìm kiếm phần tử

34

- Bước 1: Bắt đầu từ gốc
- Bước 2: So sánh dữ liệu (khóa) cần tìm với dữ liệu (khóa) của node hiện hành.
 - ▣ Nếu bằng nhau => Tìm thấy. Kết thúc
 - ▣ Nếu nhỏ hơn => Đi qua nhánh trái, Tiếp bước 2.
 - ▣ Nếu lớn hơn => Đi qua nhánh phải, Tiếp bước 2.
- Bước 3: Không thể đi tiếp nữa => Không tìm thấy. Kết thúc.

Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Xóa phần tử

35

- Tìm đến node chứa dữ liệu (khóa) cần xóa.
- Xét các trường hợp:
 - ▣ Node lá
 - ▣ Node chỉ có 1 con
 - ▣ Node có 2 con: dùng phần tử thế mạng để xóa thế.

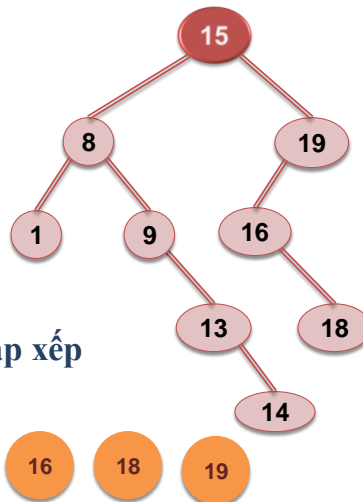
Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Sắp xếp

36

- Cho cây nhị phân tìm kiếm
- Thứ tự duyệt các node nếu sử dụng Duyệt giữa?
- Nêu nhận xét

- ▣ Có thể dễ dàng tạo dữ liệu sắp xếp nếu dùng phép duyệt giữa

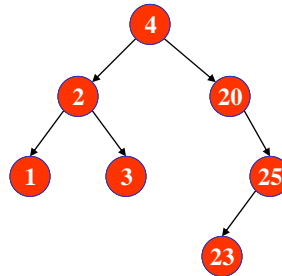


Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Phép duyệt cây

37

◉ Duyệt trước

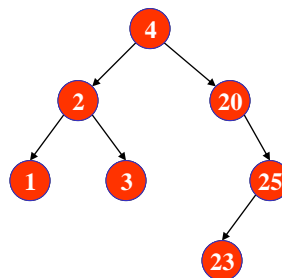


Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Phép duyệt cây

38

◉ Duyệt giữa

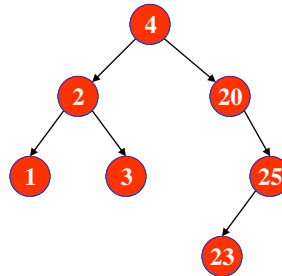


Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Phép duyệt cây

39

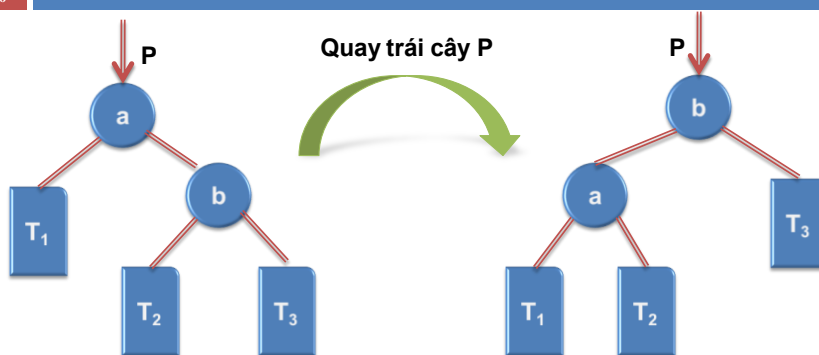
◉ Duyệt sau



Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Phép quay trái

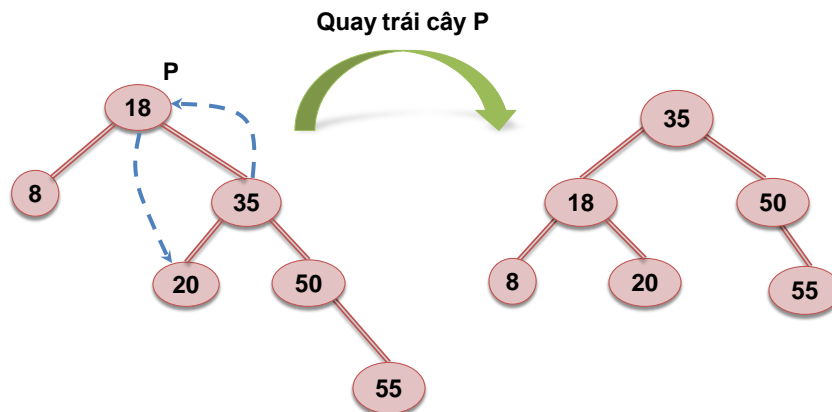
40



Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Phép quay trái

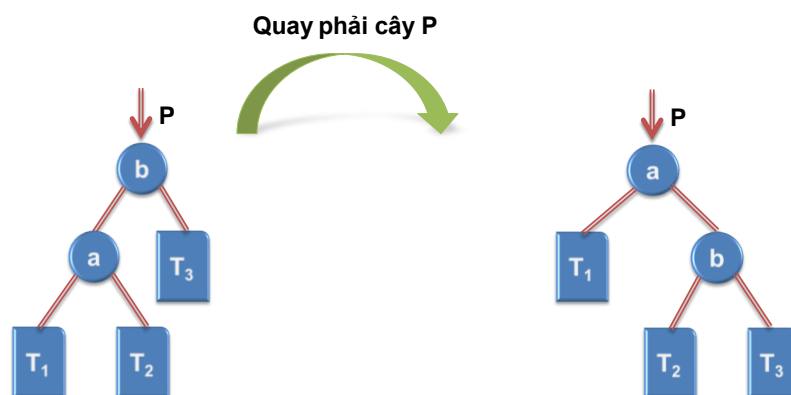
41



Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Phép quay phải

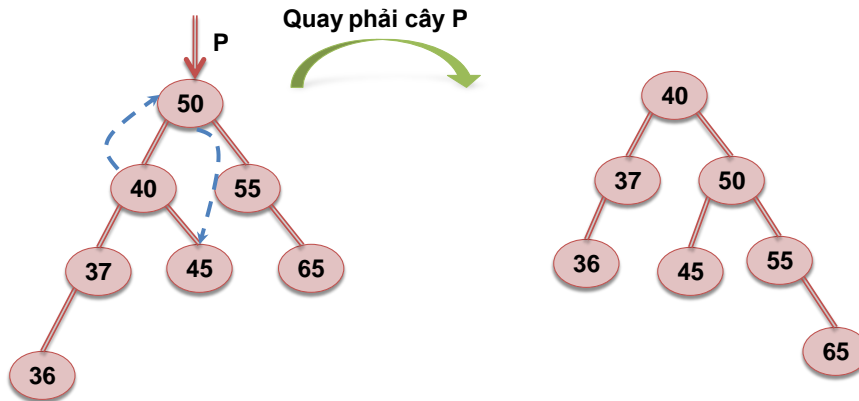
42



Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Phép quay phải

43



Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Thời gian thực hiện các phép toán

44

- Đối với phép tìm kiếm:
 - ▣ Trường hợp tốt nhất: mỗi nút (trừ nút lá) đều có 2 con:
 $O(\log_2 n)$ (chính là chiều cao của cây).
 - ▣ Trường hợp xấu nhất: cây trở thành danh sách liên kết:
 $O(n)$.
 - ▣ Trường hợp trung bình là bao nhiêu?
 $O(\log_2 n)$

Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Ví dụ

45

- ◉ Tạo cây nhị phân tìm kiếm theo thứ tự nhập như sau: 1, 8, 9, 12, 14, 15, 16, 18, 19

Cấu trúc dữ liệu và giải thuật - HCMUS 2011

Ví dụ

46

- ◉ Tạo cây nhị phân tìm kiếm theo thứ tự nhập như sau: 1, 8, 9, 12, 14, 15, 16, 18, 19



Cấu trúc dữ liệu và giải thuật - HCMUS 2011