CÂY VÀ CÂY NHỊ PHÂN (TREE AND BINARY TREE)

DATA STRUCTURES AND ALGORITHMS

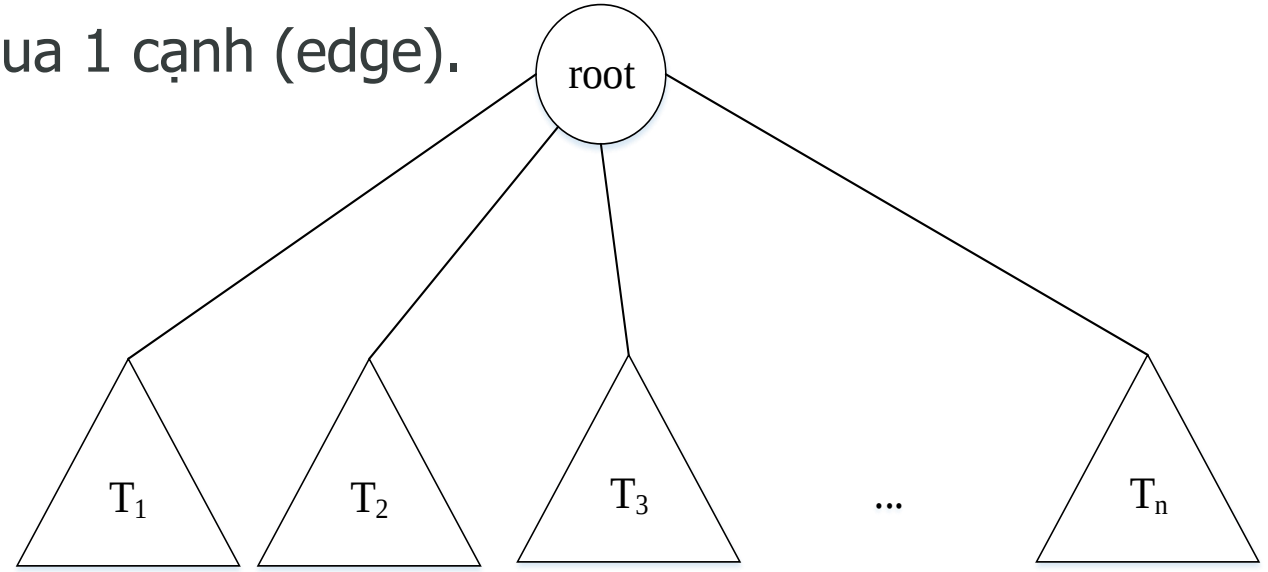
ThS Nguyễn Thị Ngọc Diễm
diemntn@uit.edu.vn

Nội dung

- Cây (Tree)
- Cây nhị phân (Binary Tree)

Định nghĩa Cây (Tree)

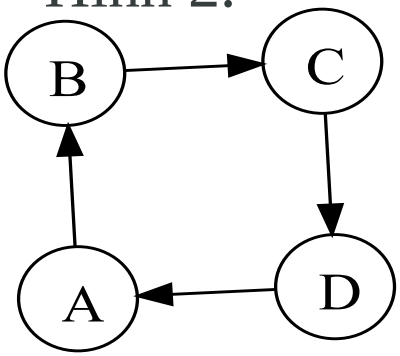
- Cây là một tập hợp T các phần tử (gọi là nút hay node):
  - Tập hợp T có thể rỗng
  - (Định nghĩa đệ quy) Nếu cây không rỗng, có một nút đặc biệt gọi là nút gốc (root), các nút còn lại được chia thành tập các cây con (subtree) T_1, T_2, …,T_n. Các cây con này liên kết trực tiếp với node gốc thông qua 1 cạnh (edge).



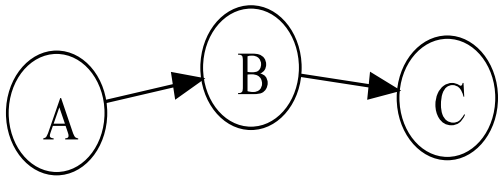
Câu hỏi

- Hình nào sau đây không phải là cây?

Hình 2.



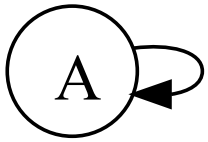
Hình 4.



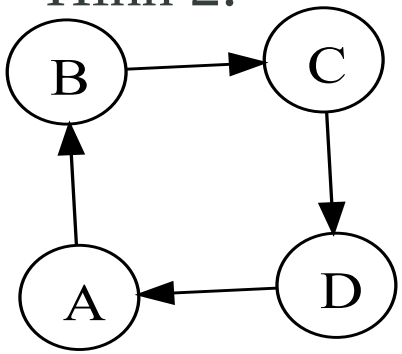
Câu hỏi

- Hình nào sau đây không phải là cây?

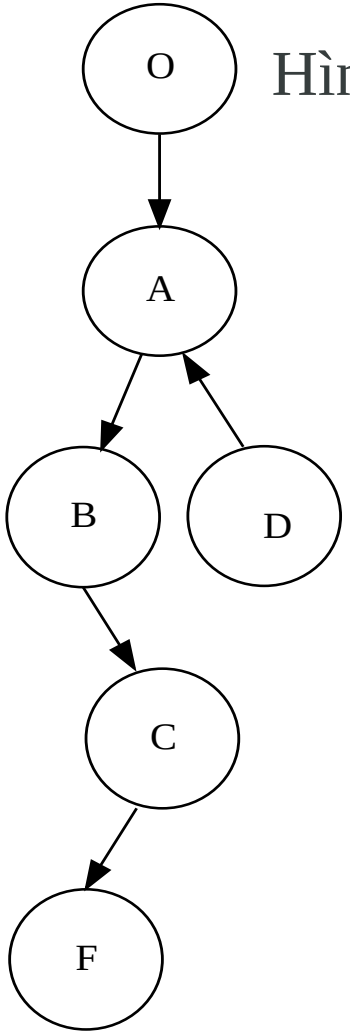
Hình 1.



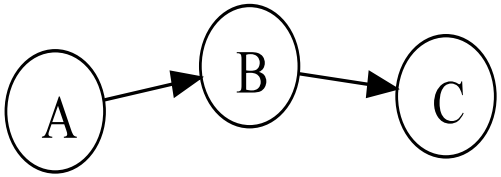
Hình 2.



Hình 3.



Hình 4.



Một số Khái niệm

- *Bậc của một nút (degree of node)*
  - Là số cây con của node đó
- *Bậc của một cây (degree of tree)*
  - Là bậc lớn nhất của các node trong cây
- *Cha và con (parent and child)*
  - Mỗi node trừ node gốc đều có duy nhất 1 node cha
  - Một node có thể có số lượng node con tùy ý
  - Node A là node cha của node B khi node A ở mức i và node B ở mức i+1. Đồng thời có một cạnh nối giữa node A và B (ta còn gọi B là con của A).
- *Lá (leaves)*
  - Node không có con
- *Họ hàng (siblings)*
  - Node có cùng cha

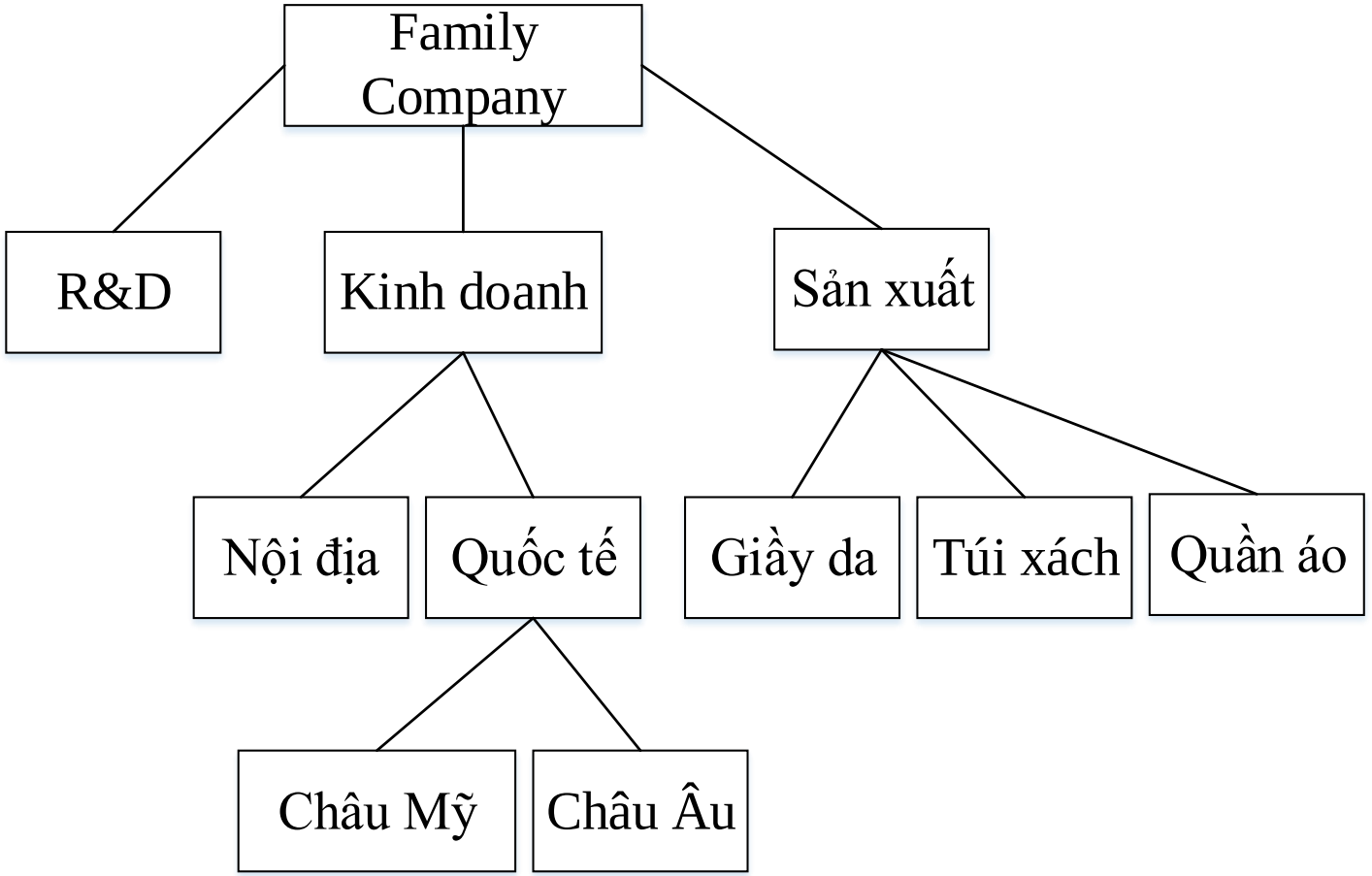
Một số Khái niệm

- *Đường đi (Path)*
  - Đường đi từ node n_1 tới n_k được định nghĩa là: Một tập các node $n_1, n_2, ..., n_k$ sao cho n_i là cha của n_{i+i} $(1 \leq i < k)$
- *Chiều dài đường đi (path length)*
  - Số lượng cạnh trên đường đi
- *Độ sâu của node (Depth/Level of node)*
  - Độ dài đường đi (path length) duy nhất từ node gốc tới node đó
  - Độ sâu của cây (The depth of a tree) bằng với chiều sâu của node lá sâu nhất.

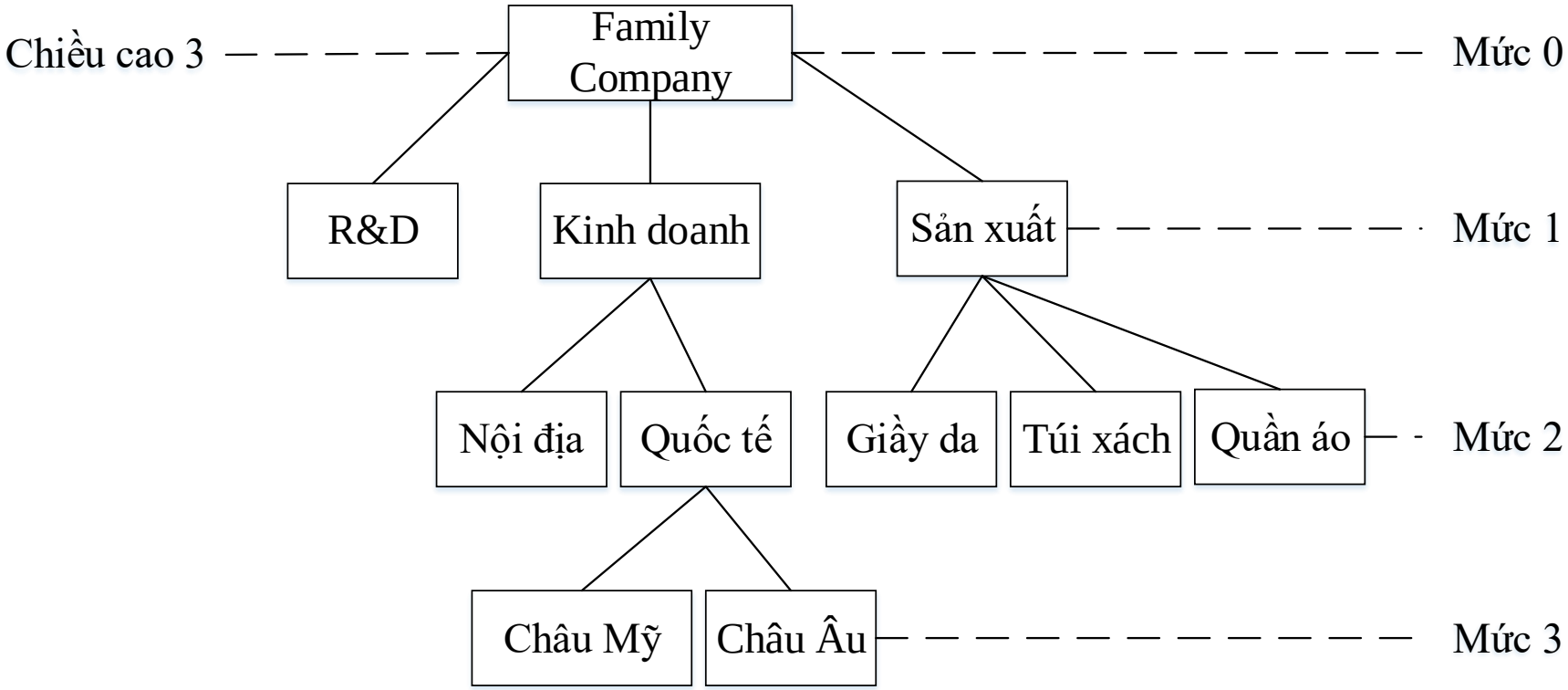
Một số Khái niệm

- *Chiều cao của node (Height of node)*
  - Chiều cao của một node bất kỳ trong cây là chiều dài dài nhất của đường đi từ node đó tới một node lá
  - Các node lá có chiều cao bằng 0
- *Chiều cao của cây (Height of tree)*
  - Bằng chiều cao của node gốc
- *Tổ tiên và hậu duệ (Ancestor and descendant)*
  - Nếu có một đường nối từ nút A đến nút B và mức của nút A < mức của nút B thì ta nói A là cha ông (tiền bối) của B và B gọi là con cháu (hậu duệ) của A.

Ví dụ 1 Tổ chức dạng cây



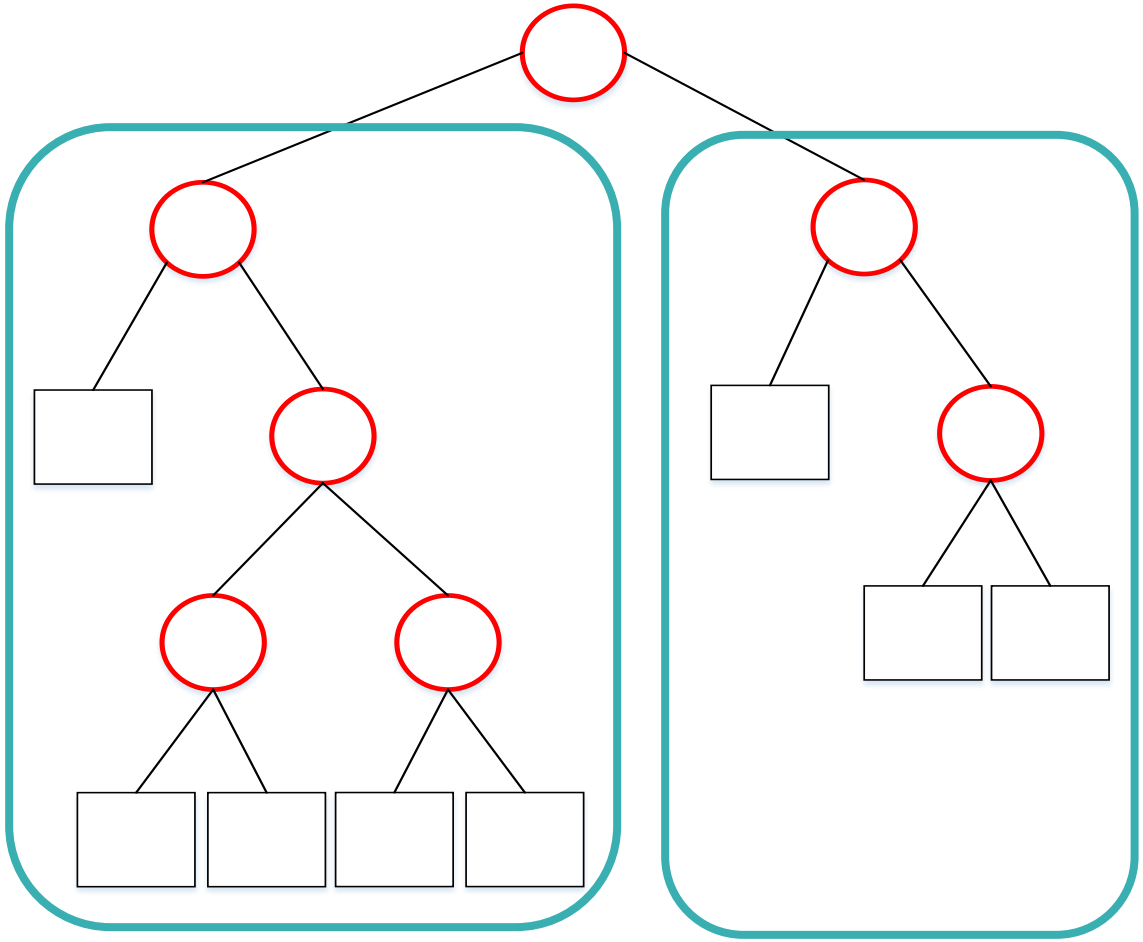
Ví dụ 1 Tổ chức dạng cây



Chiều cao node Family Company: 3
Chiều cao node R&D: 0
Chiều cao node Kinh doanh: 2
Chiều cao node Sản xuất: 1
Chiều cao node Nội địa: 0
Chiều cao node Châu Âu: 0

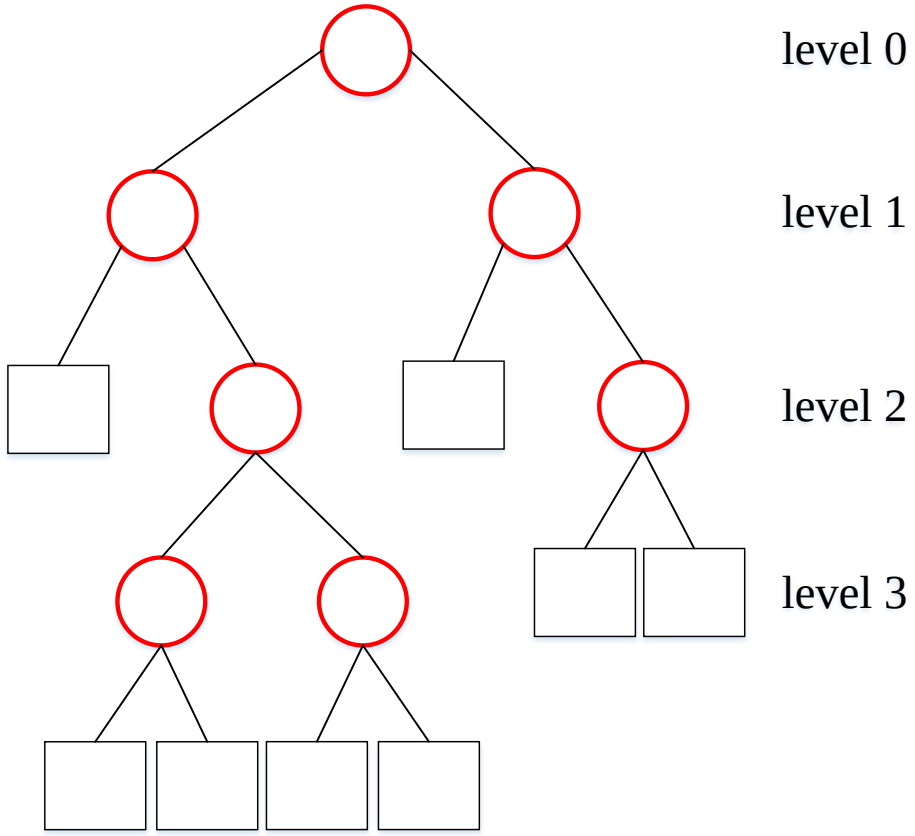
Cây Nhị phân (Binary Tree)

- Mỗi node có tối đa 2 cây con



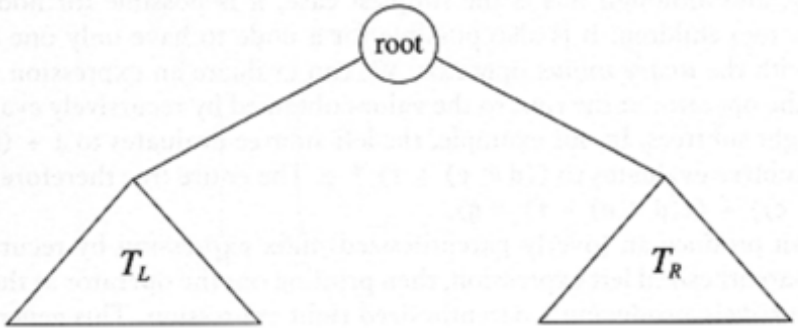
Một số tính chất của cây Nhị phân

- Số node nằm ở mức i $\leq 2^i$.
- Số nút lá $\leq 2^h$, với h là chiều cao của cây.
- Cây nhị phân có chiều cao h ($h \geq 0$) sẽ có tối đa: $2^{h+1} - 1$ node
- Chiều cao của cây $h \geq \log_2(N)$ với N = số nút trong cây
- Số lượng node nhiều nhất trên mức thứ k ($k \geq 0$, gốc của cây nằm ở mức 0) của cây nhị phân là: 2^k.

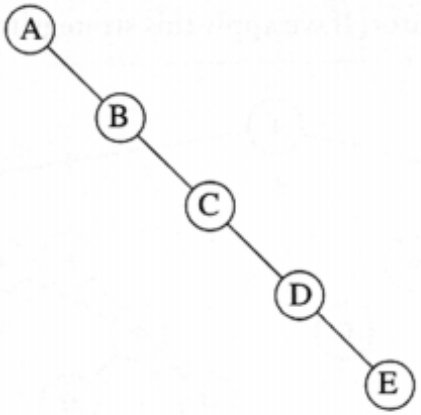


Cây nhị phân

- A tree in which no node can have more than two children



- The depth of an "average" binary tree is considerably smaller than N, eventhough in the worst case, the depth can be as large as N – 1.

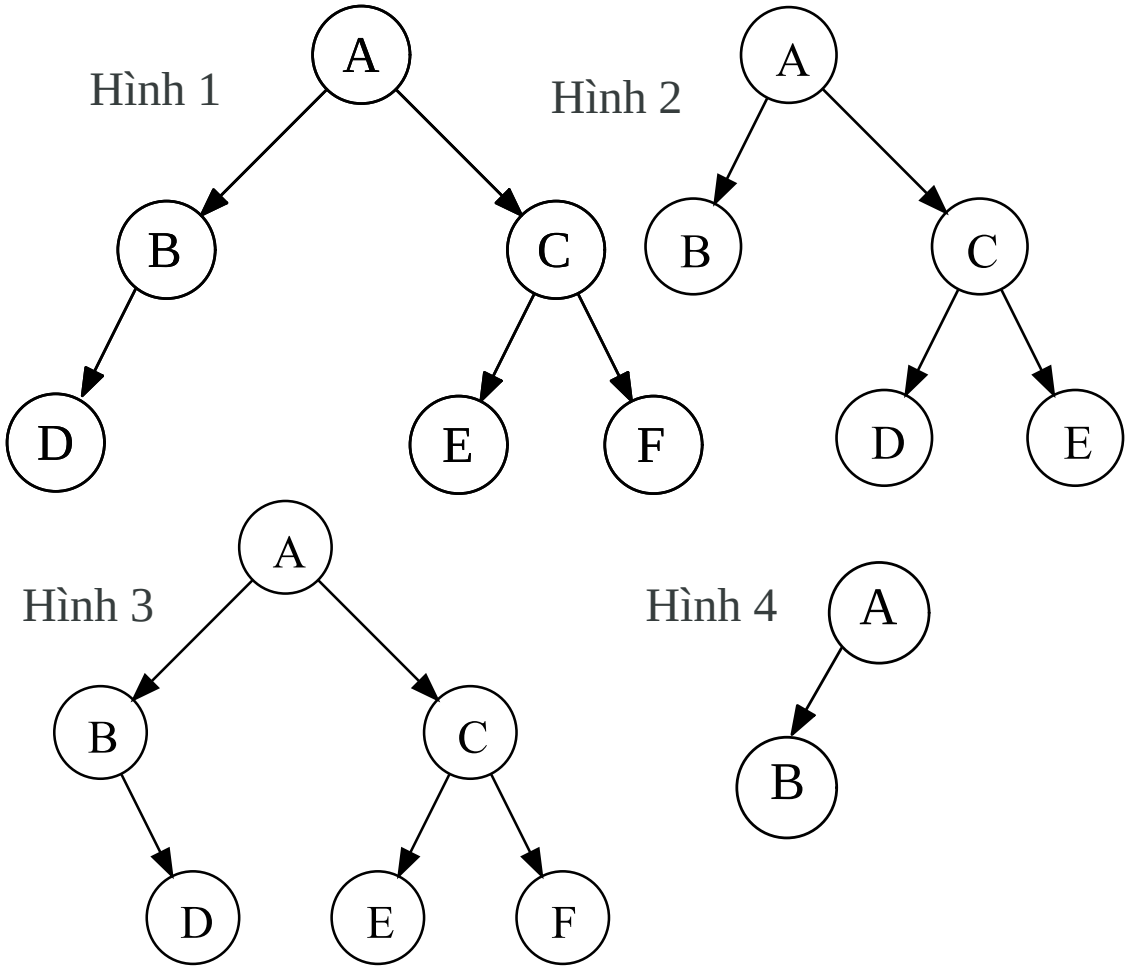


Cây nhị phân hoàn chỉnh (complete binary tree)

- Cây nhị phân hoàn chỉnh (complete binary tree) là: Cây nhị phân nếu không tính đến độ sâu cuối cùng thì hoàn toàn chứa đầy đủ các node, và tất cả các node ở độ sâu cuối cùng sẽ lệch sang trái nhất có thể.

Cây nhị phân hoàn chỉnh (complete binary tree)

- Cây nào trong những cây sau đây là cây nhị phân hoàn chỉnh (complete binary tree)?



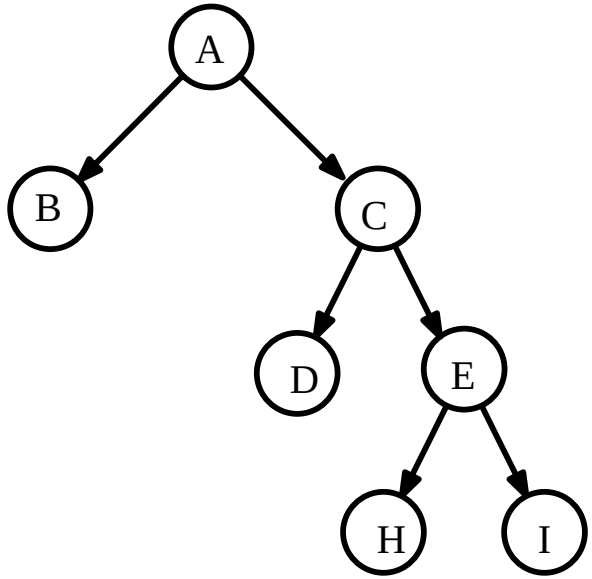
Cây nhị phân đầy đủ (full binary tree)

- Cây nhị phân đầy đủ (full binary tree): Là Cây nhị phân mà mỗi node không phải node lá thì sẽ có chính xác 2 node con

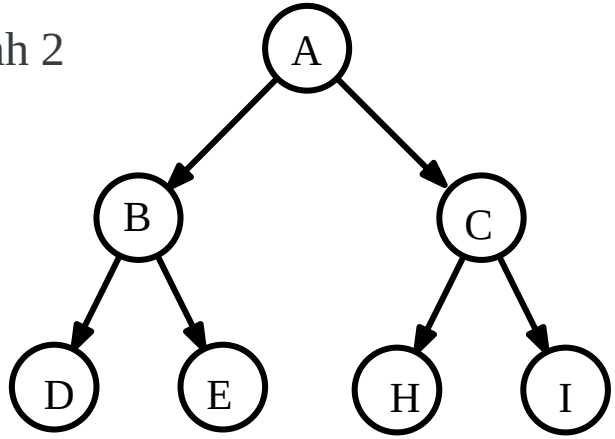
Cây nhị phân đầy đủ (full binary tree)

- Cây nào trong những cây sau đây là cây nhị phân đầy đủ?

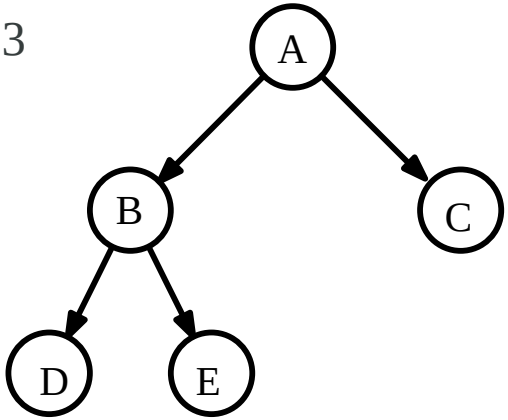
Hình 1



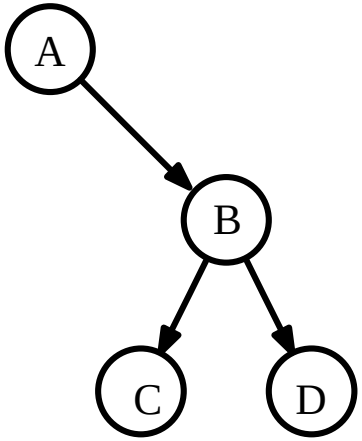
Hình 2



Hình 3

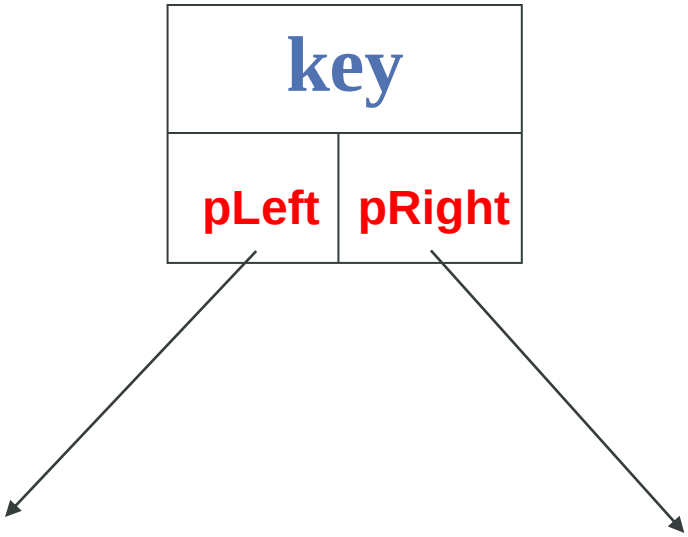


Hình 4



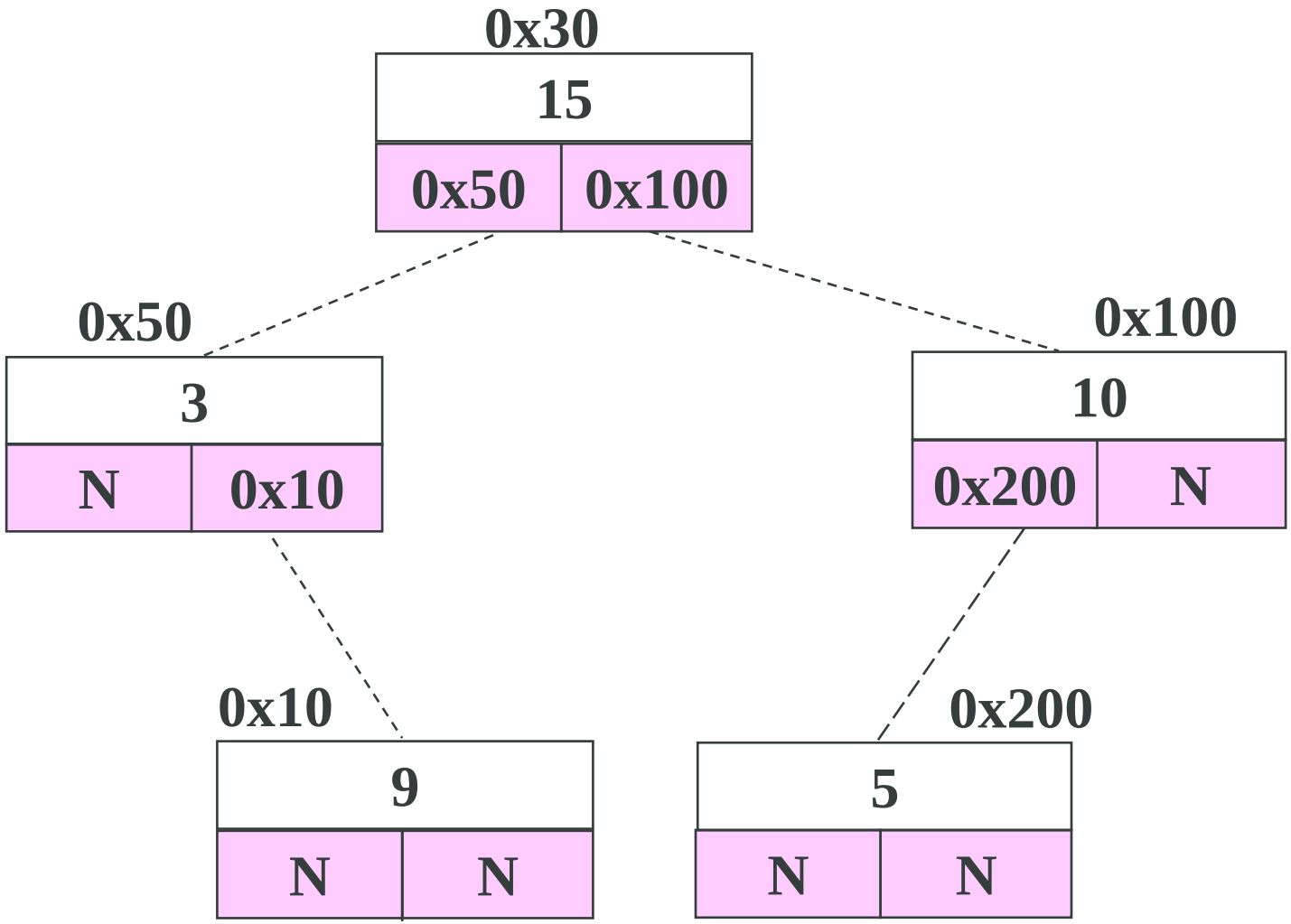
Cấu trúc dữ liệu của Cây Nhị phân

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

typedef TNode *TREE;

// Khai báo cây T, T1
TREE T, T1; // hay: TNODE* T, T1;
```



Ví dụ Cây được tổ chức trong bộ nhớ trong

- Cây nhị phân lưu trữ danh sách các số sau: 15, 3, 10, 9, 5



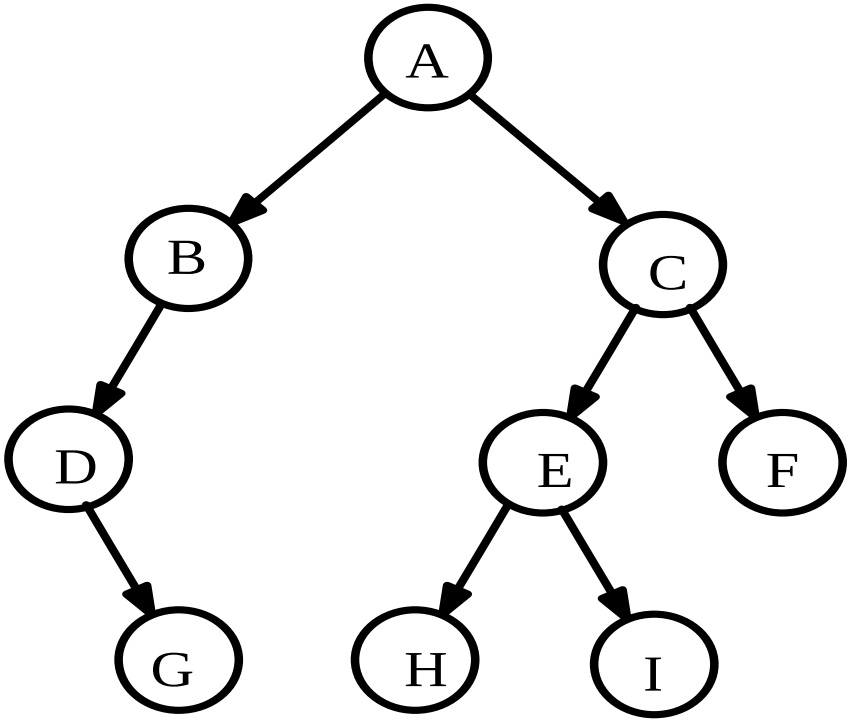
Duyệt cây Nhị phân

- Có 3 trình tự thăm gốc: (3 cách cơ bản để duyệt cây)
  - Duyệt trước (preorder)
  - Duyệt giữa (inorder)
  - Duyệt sau (postorder)
- Độ phức tạp O ($log_2(h)$)
  - Trong đó h là chiều cao cây

Ví dụ Kết quả của phép duyệt cây

- In cây sau theo thứ tự preorder, inorder và postorder:

- preorder: A B D G C E H I F
- inorder: G D B A H E I C F
- Postorder: G D B H I E F C A



Duyệt trước – preorder (NLR)

```
void preorder(TREE Root) {
   if (Root != NULL) {
        <Xử lý Root>; //Xử lý tương ứng theo nhu cầu
        preorder(Root->pLeft);
        preorder(Root->pRight);
   }
}
```

Duyệt giữa - inorder (LNR)

```
void inorder(TREE Root) {
   if (Root != NULL) {
        inorder(Root->pLeft);
       <Xử lý Root>; //Xử lý tương ứng theo nhu cầu
        inorder(Root->pRight);
   }
}
```

Duyệt sau - postorder (LRN)

```
void postorder(TREE Root) {
   if (Root != NULL) {
        postorder(Root->pLeft);
        postorder(Root->pRight);
        <Xử lý Root>; //Xử lý tương ứng theo nhu cầu
   }
}
```

Example: Expression Trees

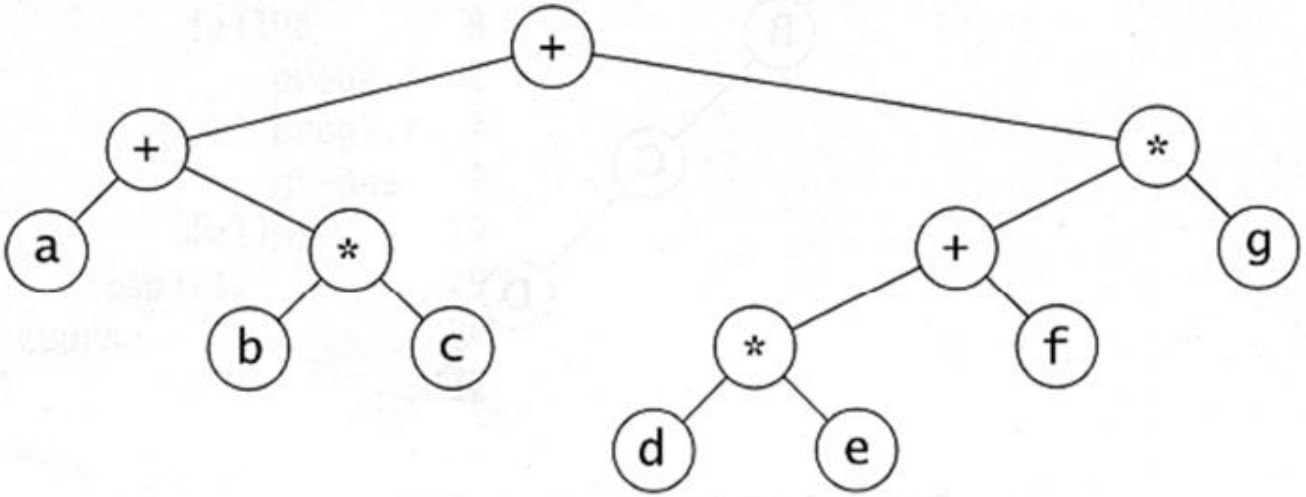


Figure 4.14 Expression tree for (a + b * c) + ((d * e + f) * g)

- Leaves are operands (constants or variables)
- The other nodes (internal nodes) contain operators
- Will not be a binary tree if some operators are not binary

Preorder, Postorder and Inorder

- Preorder traversal
  - node, left, right (NLR)
  - prefix expression
    - ++a*bc*+*defg

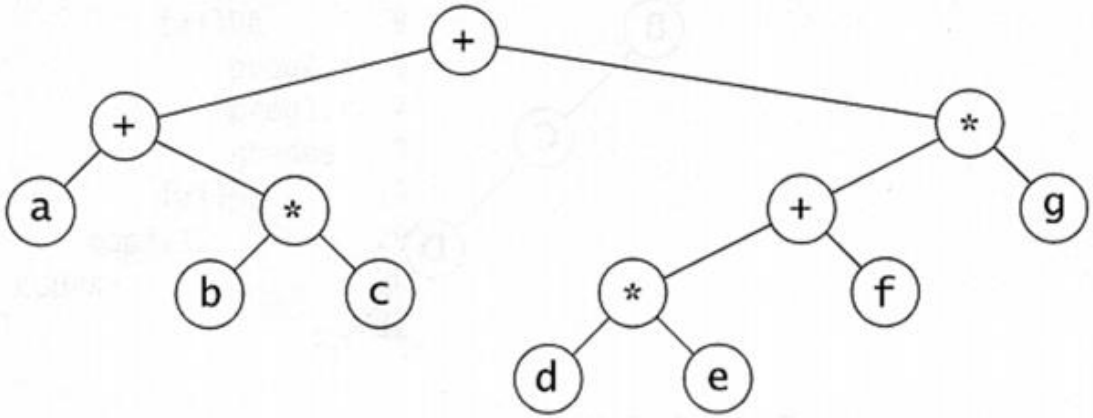


Figure 4.14 Expression tree for (a + b * c) + ((d * e + f) * g)

Preorder, Postorder and Inorder

- Postorder traversal
  - left, right, node (LRN)
  - postfix expression
    - abc*+de*f+g*+
- Inorder traversal
  - left, node, right (LNR)
  - infix expression
    - a+b*c+d*e+f*g

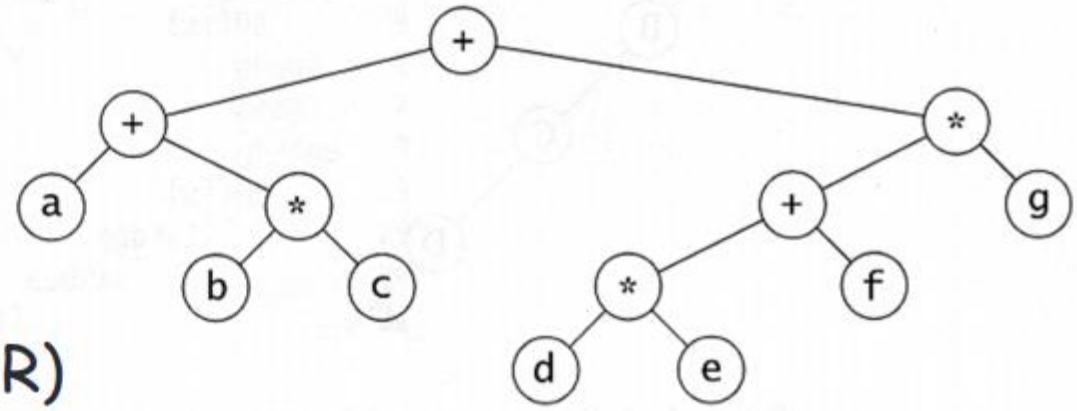


Figure 4.14 Expression tree for (a + b * c) + ((d * e + f) * g)

Cách biểu diễn cây nhị phân khác

- Đôi khi còn quan tâm đến cả quan hệ 2 chiều cha con chứ không chỉ một chiều như định nghĩa. Cấu trúc cây nhị phân như sau:

```
struct TNode {
    DataType Key;
    TNode* pParent;
    TNode* pLeft;
    TNode* pRight;
};
typedef TNode* Tree;
```

- Cây nhị phân tìm kiếm giúp dễ dàng trong tìm kiếm thông tin.

Câu hỏi và Bài tập

1. Định nghĩa cây và cấu trúc cây.
2. Nêu một số tính chất của cây.
3. Nêu định nghĩa và một số ứng dụng của cây nhị phân.
4. Nêu một số tính chất của cây nhị phân. Cấu trúc biểu diễn cây nhị phân như thế nào?
5. Trình bày các kiểu duyệt cây nhị phân.

Chúc các em học tốt!