

# 자료구조 Data Structure | 조행래

## 트리

트리와 이진트리의 개념

## 학습 목표

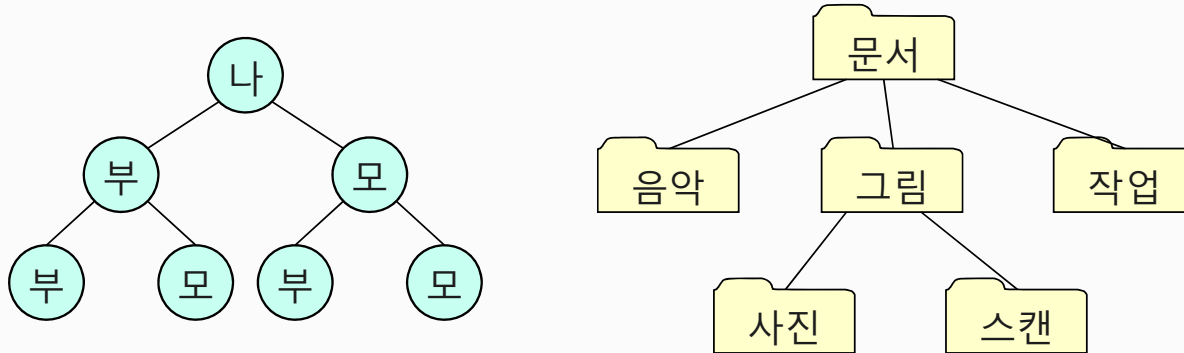
---

- 트리의 개념을 이해하고, 트리에서 사용되는 용어들을 정의할 수 있다.
- 이진 트리를 정의할 수 있고, 이진 트리를 표현하는 방법들을 설명할 수 있다.

# 1. 트리의 개념

## ■ 트리란?

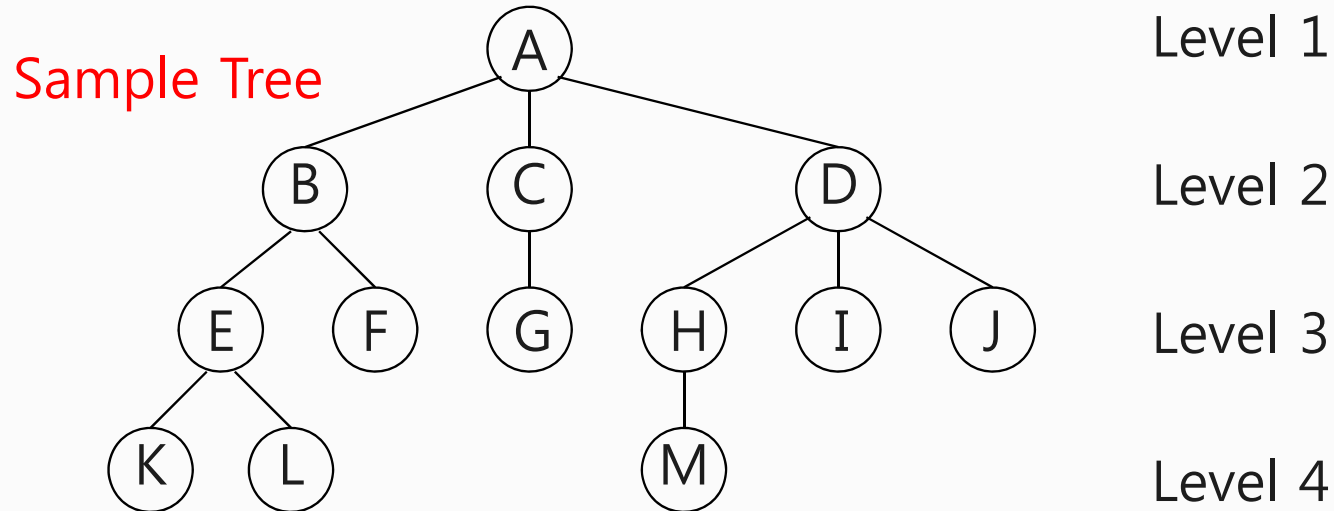
- 계층적 구조의 자료를 표현할 때 사용
  - 가계도, 회사의 조직, 폴더 구조 등



## ■ 트리의 정의

- 하나 이상의 노드로 이루어진 유한집합
- Root라고 하는 노드가 하나 존재
- 나머지 노드들은  $n$  ( $\geq 0$ )개의 집합  $T_1, \dots, T_n$ 으로 분할
  - $T_1, \dots, T_n$ : 분리된 트리 (Root의 서브 트리)

## 2. 트리에 관련된 용어들

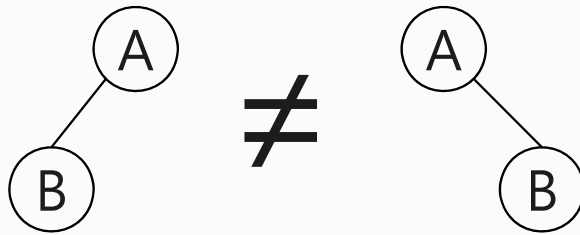


- 노드의 차수(degree) (A: 3), 트리의 차수(degree) (3)
- 단말 노드(leaf or terminal node) (K, L, F, G, M, I, J)
- 부모 노드(E: B), 자식 노드(B: E & F), 형제 노드(E & F)
- 조상 노드(M: H, D, A), 자손 노드(B: E, F, K, L)
- Level (Root: 1), 높이 또는 깊이(4)

### 3. 이진 트리의 개념

- 이진 트리(binary tree)란?

- 모든 노드의 차수(degree)가 2를 넘지 않는 트리
  - 일반 트리에 비해 구현이 용이
- 왼쪽 서브 트리와 오른쪽 서브 트리가 구분



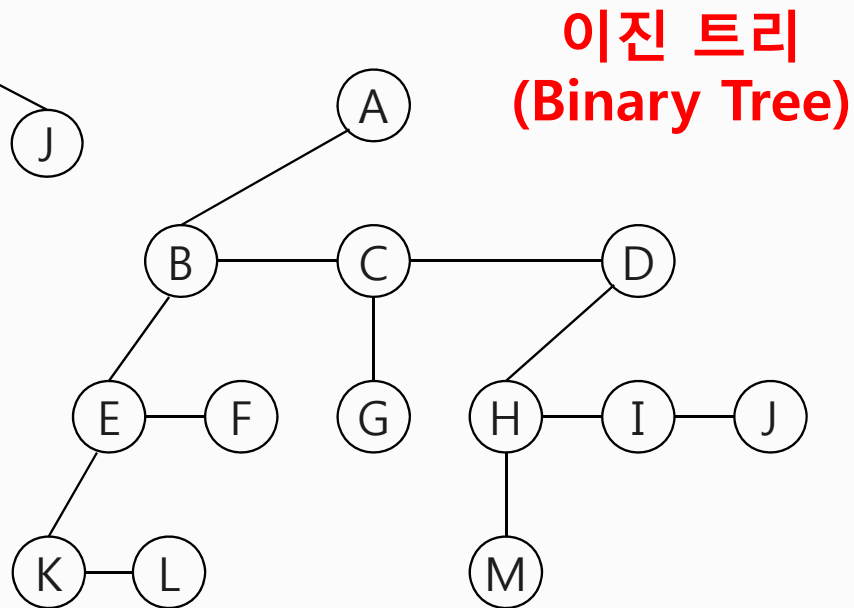
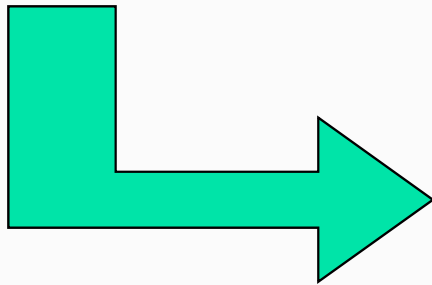
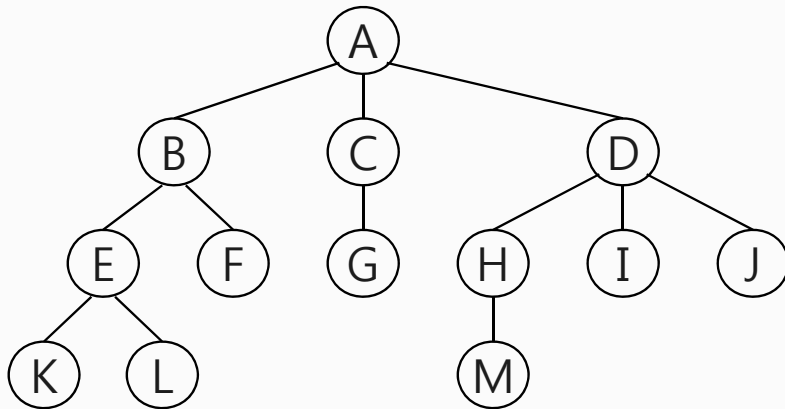
- 이진 트리의 정의

- 유한 개의 노드들의 집합으로서
- 노드 수는 0이 될 수 있으며,
- 하나의 root 노드와 왼쪽 서브 트리, 그리고 오른쪽 서브 트리로 구성
- 각 서브 트리는 다시 이진 트리이다.

# 트리를 이진 트리로 변환하는 방법

## ■ Left Child - Right Sibling 표현

- 노드 A의 제일 왼쪽 노드 → A의 왼쪽 자식 노드
- A의 나머지 자식 노드들 → 자식 노드의 오른쪽 자식



이진 트리  
(Binary Tree)

## 4. 이진 트리의 성질

### ■ 최대 노드 수

- 이진 트리의 레벨  $i$ 에서 최대 노드 수는  $2^{i-1}$  ( $i \geq 1$ )
- 깊이가  $k$ 인 이진 트리가 가질 수 있는 최대 노드 수 =

$$\sum_{i=1}^k 2^{i-1} = 2^k - 1$$

### ■ 단말 노드 수와 차수가 2인 노드 수

- $n_0$ : 단말 노드 수,  $n_2$ : 차수가 2인 노드의 수

$$\Rightarrow n_0 = n_2 + 1$$

$$(\text{증명}) \ n = n_0 + n_1 + n_2 \ \& \ n = E + 1 = n_1 + 2n_2 + 1$$

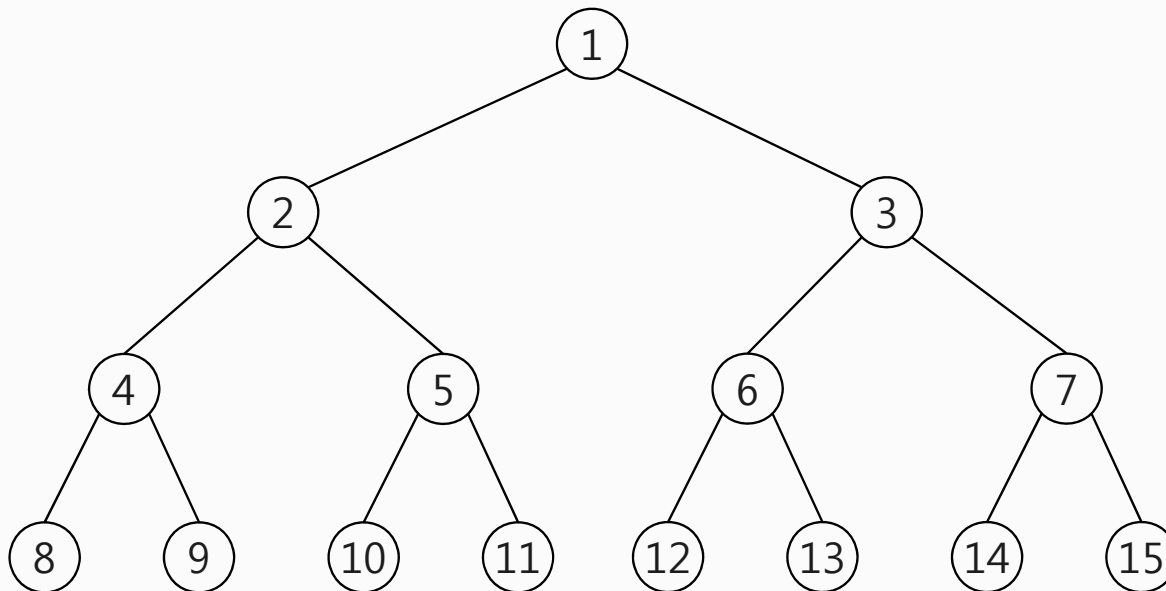
### ■ 정의: 깊이가 $k$ 인 포화 이진 트리(full binary tree)

- 깊이가  $k$ 이고 노드 수가  $2^k - 1$  ( $k \geq 0$ ) 인 이진 트리

# 완전 이진 트리(Complete Binary Tree)

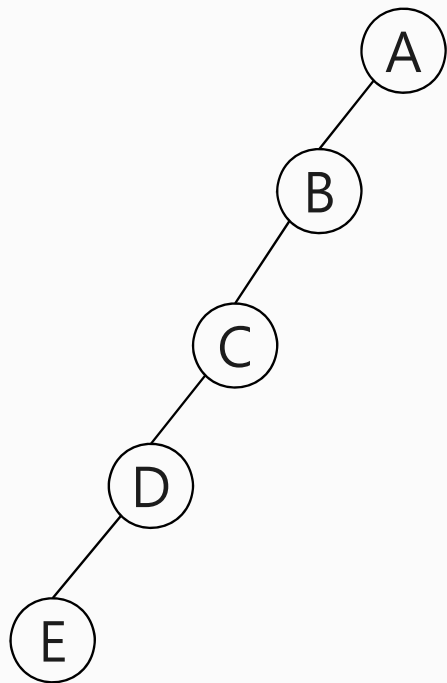
## ■ 완전 이진 트리란?

- 깊이가  $k$ 이고 노드 수가  $n$ 인 이진 트리의 각 노드들이 깊이가  $k$ 인 포화 이진 트리에서 1부터  $n$ 까지의 번호를 붙인 노드들과 1대 1로 일치하는 이진 트리

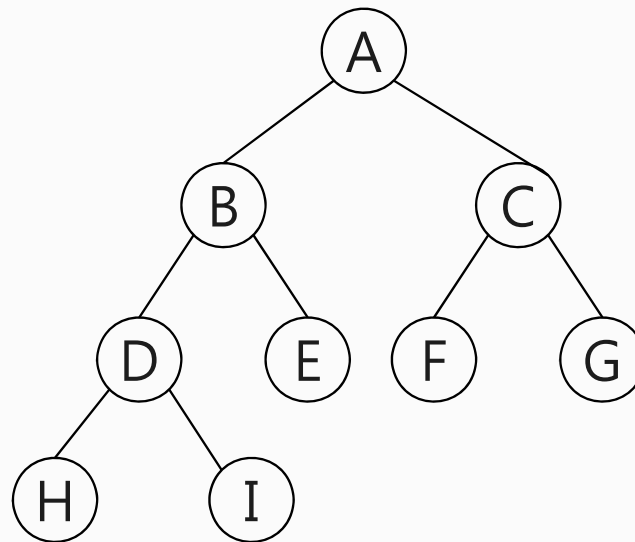




## 이진 트리의 예



편향(skewed)  
이진 트리



완전(complete)  
이진 트리

Level  
1  
2  
3  
4  
5

## 5. 이진 트리의 표현

- **배열 표현법** – 1차원 배열에 저장
  - 루트 노드는 [1]에 저장
  - 부모 노드가 [i]에 저장될 경우,
    - 왼쪽 자식 노드는  $[2*i]$ 에 저장
    - 오른쪽 자식 노드는  $[2*i+1]$ 에 저장
- **배열 표현의 예**

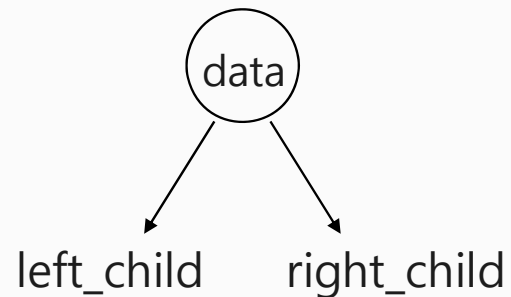
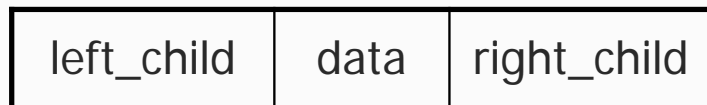
[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	...	[16]
	A	B		C				D		E

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	...
	A	B	C	D	E	F	G	H	I	

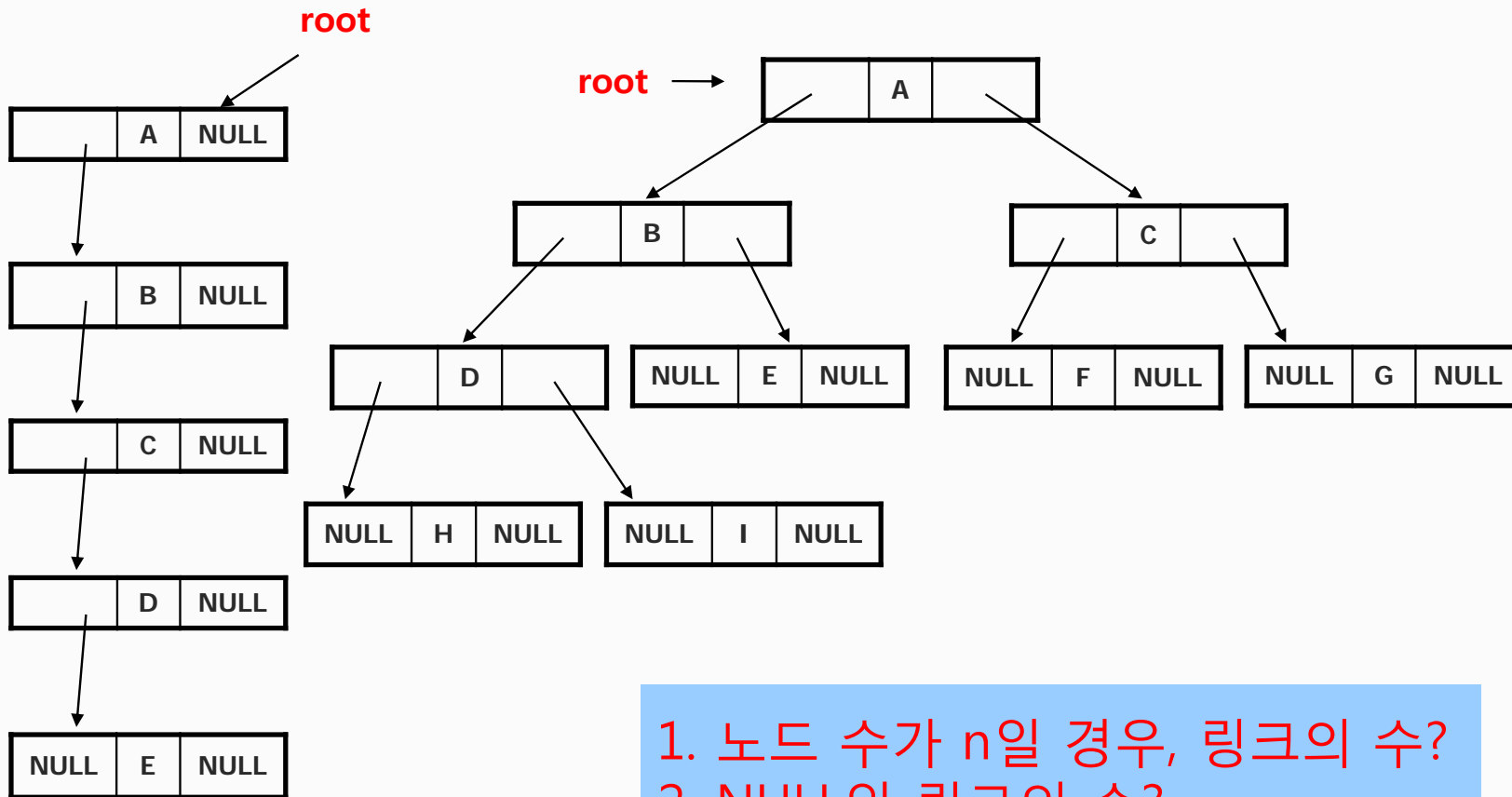
## 이진 트리의 표현(계속)

- 링크 표현법
  - 노드를 구조체로 표현

```
struct node {  
    char  data;  
    struct node *left_child;  // 왼쪽 자식  
    struct node *right_child; // 오른쪽 자식  
};
```



## 링크 표현법의 예



1. 노드 수가  $n$ 일 경우, 링크의 수?
2. NULL인 링크의 수?



## 요약 정리

- 트리의 개념 및 트리에서 사용되는 용어들을 정리
- 이진 트리의 개념과 이진 트리를 표현하는 두 가지 방법을 설명