

Hanyang Univ.

자료구조론 12주차

강진영

CONTENTS

1

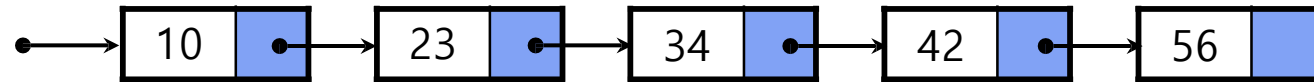
트리의 이해

2

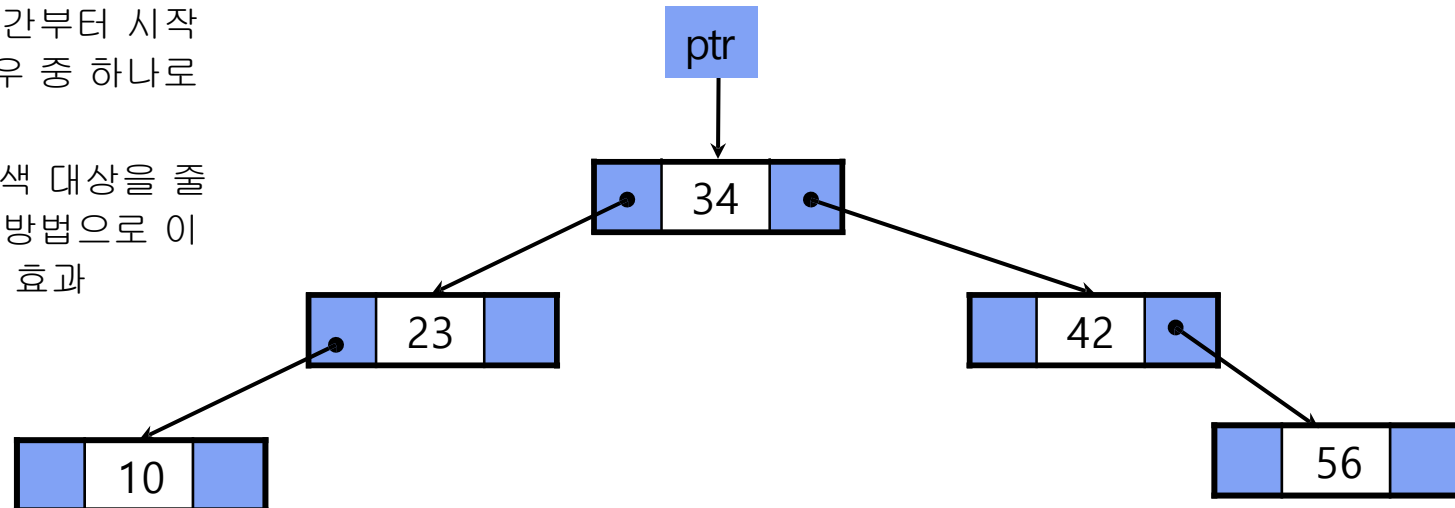
이진 트리의 이해

트리 자료구조의 필요성

- 연결리스트 삽입, 삭제 시 데이터를 이동하지 않는 장점은 유지하고 검색 시 노드의 처음부터 찾아가야 하는 단점을 보완하는 자료구조
- 데이터를 중간부터 찾아가는 이진검색의 장점을 활용
- 연결리스트의 포인터를 중간에 두는 방법론

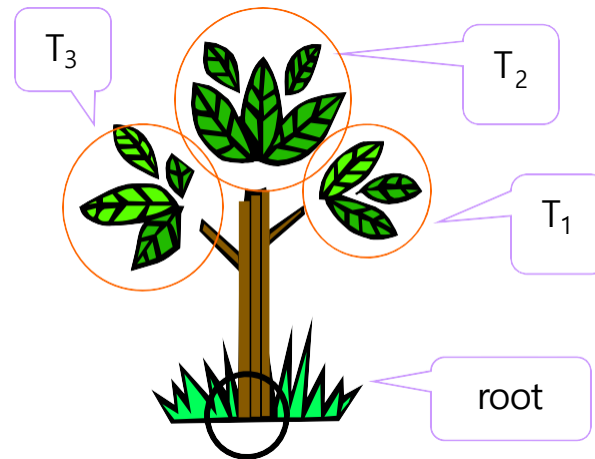


- 검색을 중간부터 시작하여 좌, 우 중 하나로 분기
- 1/2 씩 검색 대상을 줄여나가는 방법으로 이진 검색의 효과



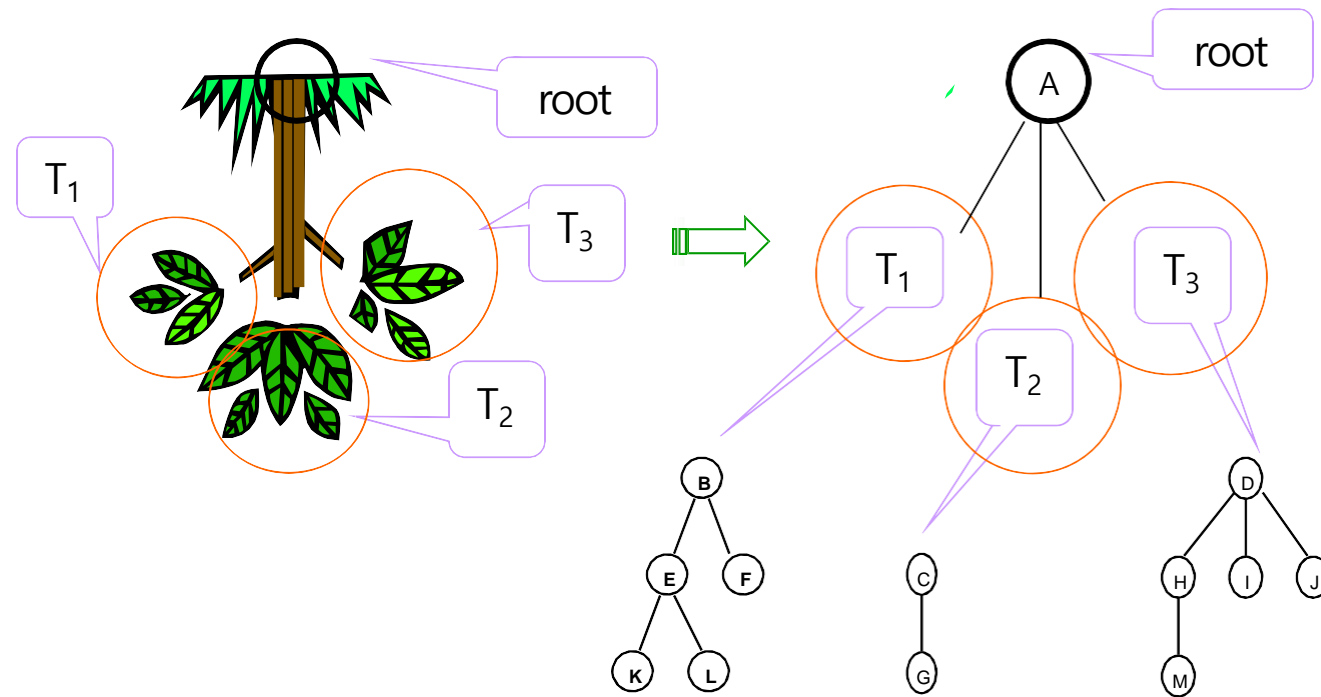
트리의 정의

- 트리는 1개 이상의 노드를 갖는 집합으로 노드들은 다음 조건을 만족
 - 1) 트리에는 루트(root)라고 부르는 특별한 노드가 존재
 - 2) 다른 노드들은 원소가 중복되지 않는 n 개의 부속 트리 (subtree) T_1, T_2, \dots, T_n 으로 나누어지며 T_i 각각은 루트의 부속 트리 (트리는 사이클이 없는 그래프(acyclic graph) 이며 계층 구조

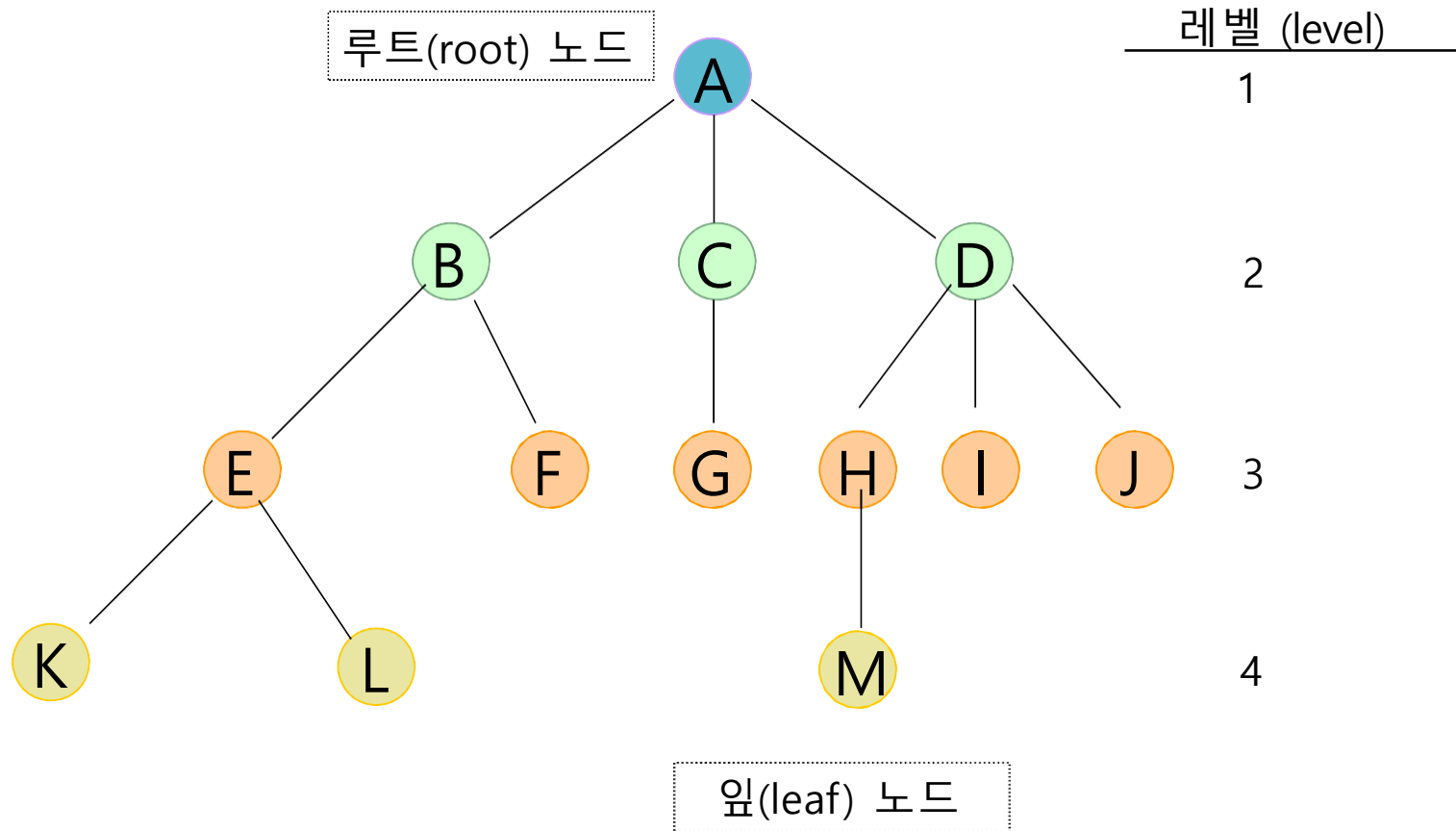


트리의 표현 (1/2)

- 자료구조 트리(Tree)는 나무를 거꾸로 표현하며 부속 트리 T_1, T_2, T_3 은 다시 트리 구조를 표현



트리의 표현 (2/2)

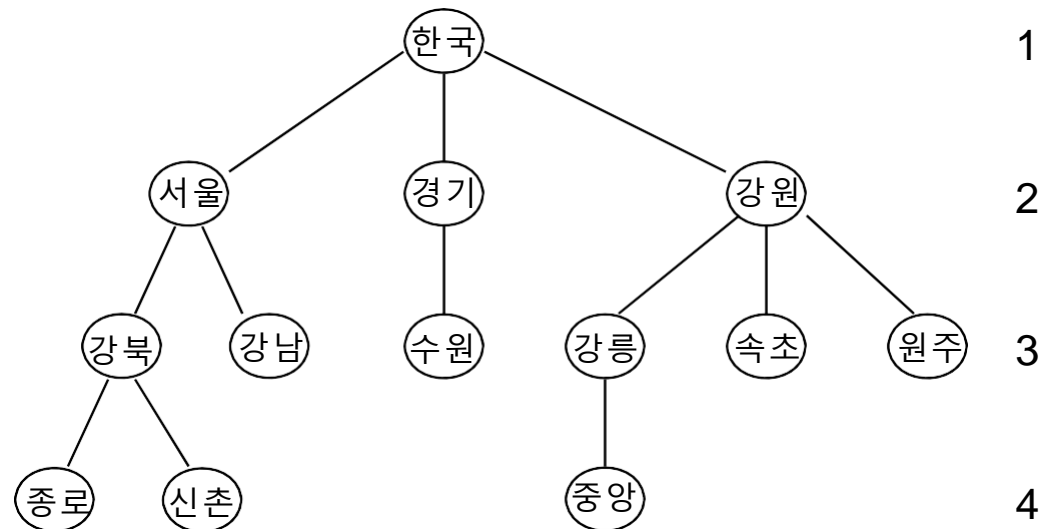


트리의 필요성

- 트리 자료구조의 필요성

➤ 트리 구조를 이용한 자료의 저장에 효율적인 경우 (예를 들어 계층적인 형태를 띄는 자료는 트리에 저장하면 자연스럽게 표현)

- 1) 회사나 정부의 조직 구조
- 2) 나라, 지방, 시·군별 계층적인 데이터
- 3) 인덱스 (인덱스는 계층적 자료 구조로서 검색에 용이)



트리 관련 용어 정리

- 트리에 관한 용어
 - 노드의 차수(degree) : 노드의 부속 트리의 개수
 - 트리의 차수(degree of tree) : 트리의 최대 차수
 - 잎(leaf, 단말, terminal) 노드 : 차수가 0인 노드, 즉 맨 끝 노드
 - 내부(internal, non-terminal) : 차수가 1 이상인 노드
 - 부모(parent) : 부속 트리(subtree)를 가진 노드
 - 자식(child) : 부모에 속하는 부속 노드
 - 형제(sibling) : 부모가 같은 자식 노드들
 - 조상(ancestor) : 노드의 부모 노드들의 총 집합
 - 자손(descendant) : 노드의 부속 트리에 있는 모든 노드들
 - 레벨(level) : 루트 노드들로부터 깊이(루트 노드의 레벨 = 1)
 - 트리의 깊이(depth of tree) : 트리에 속한 노드의 최대 레벨

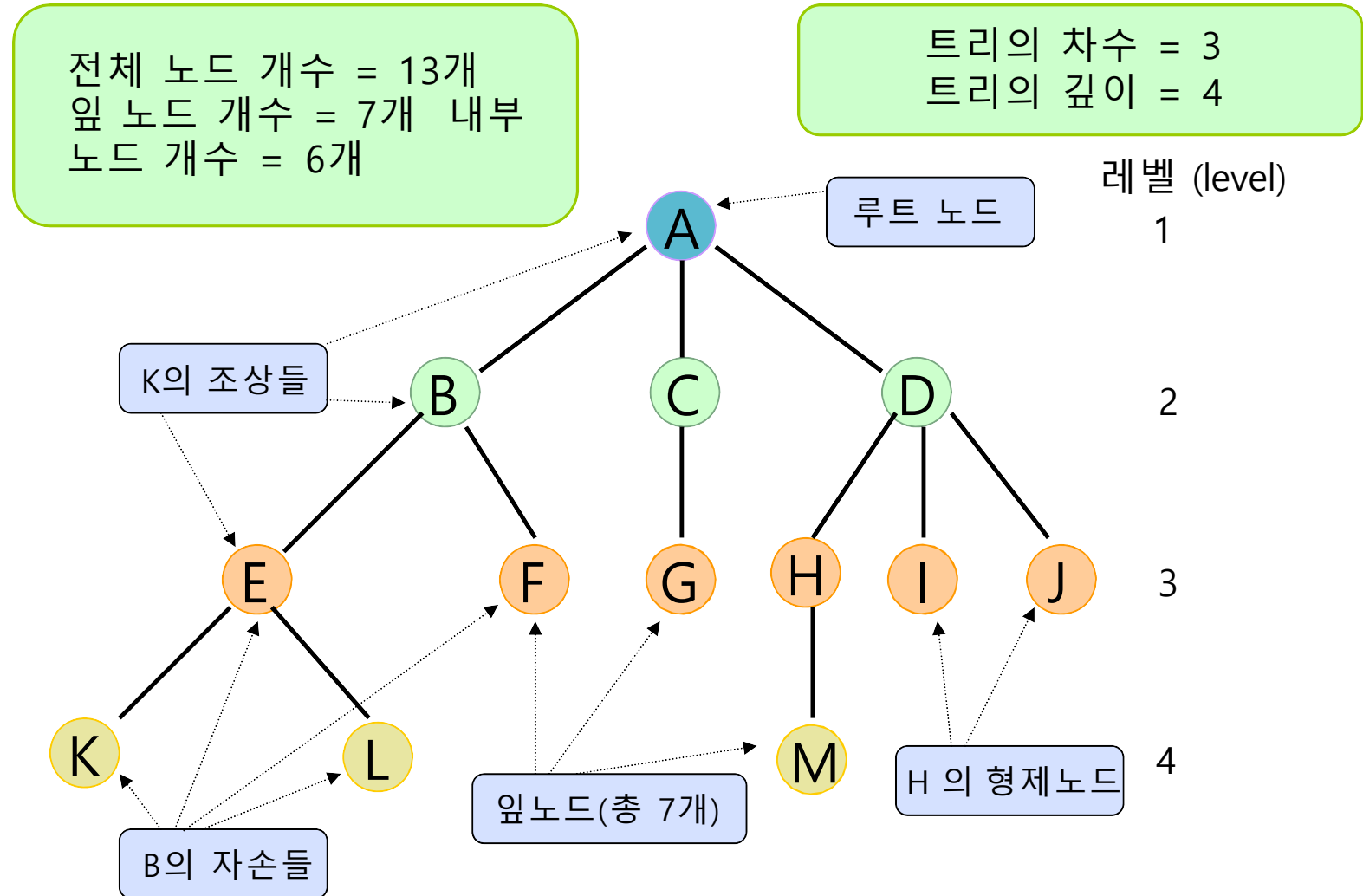
1. 트리의 이해

Contents

1. 트리의 이해

2. 이진 트리의 이해

트리 구조



트리 구조를 이용한 저장

1) N-링크 표현법

- 노드에 n 개의 링크를 두고 자식의 개수만큼 링크에 저장
- 모든 노드는 자식 노드수에 관계없이 최대 n 개의 링크를 가지며 각 링크는 부속 트리가 저장된 곳을 링크
- 차수가 n 인 경우의 노드 표현



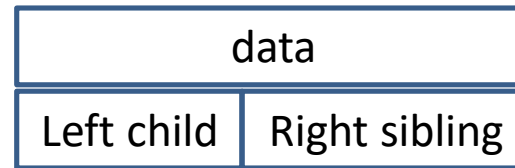
- 트리를 리스트 형태로 표현하는 경우 괄호를 사용하여 같은 레벨에 있는 노드들을 같은 괄호로 표현
- 예) $(A(B(E(K,L),F),C(G),D(H(M),I,J)))$

2) 왼쪽자식노드 - 오른쪽형제노드 표현법

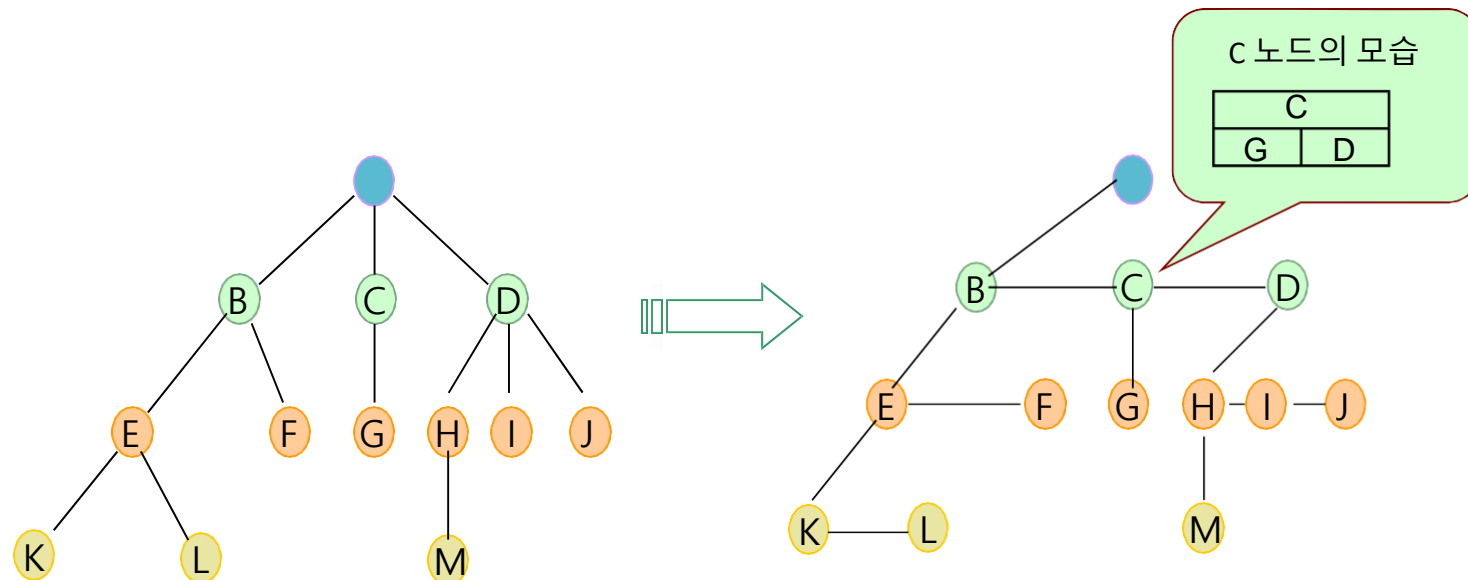
- 모든 노드가 링크를 2개 포함
- 첫째 링크는 첫번째 자식노드를 표현하고 둘째 링크는 자신의 오른쪽 형제 노드를 표현
- 노드의 길이가 2개로 고정되기 때문에 N-링크 표현 방법보다 간편

Left Child – Right Sibling

- 왼쪽자식노드 – 오른쪽형제노드 표현법

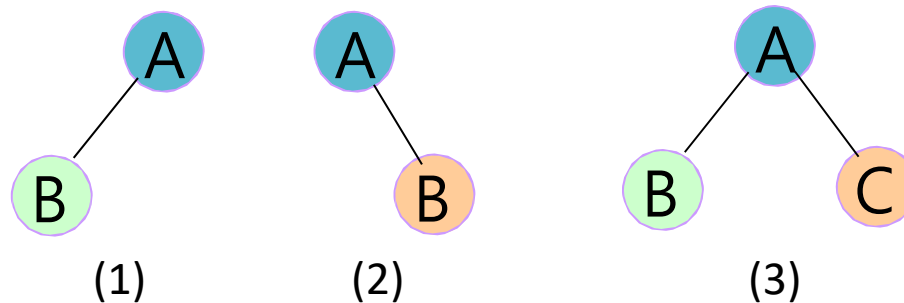


- 왼쪽자식노드 – 오른쪽형제노드 방법으로 저장된 트리



이진(Binary) 트리

- 자식 노드의 수가 2개 이하인 것을 이진 트리(binary tree)라 표현하며 응용에서는 일반 트리보다는 원래부터 이진 트리로 표현된 문제가 대부분
- 정의) 이진 트리는 유한개의 노드로 구성된 트리
 - 1) 비어있거나 혹은
 - 2) 루트 노드와 2개의 부속 트리로 구성
 - 3) 부속 트리는 왼쪽 부속 트리, 오른쪽 부속 트리로 표현
 - 4) 노드가 없는 경우도 이진 트리의 일종



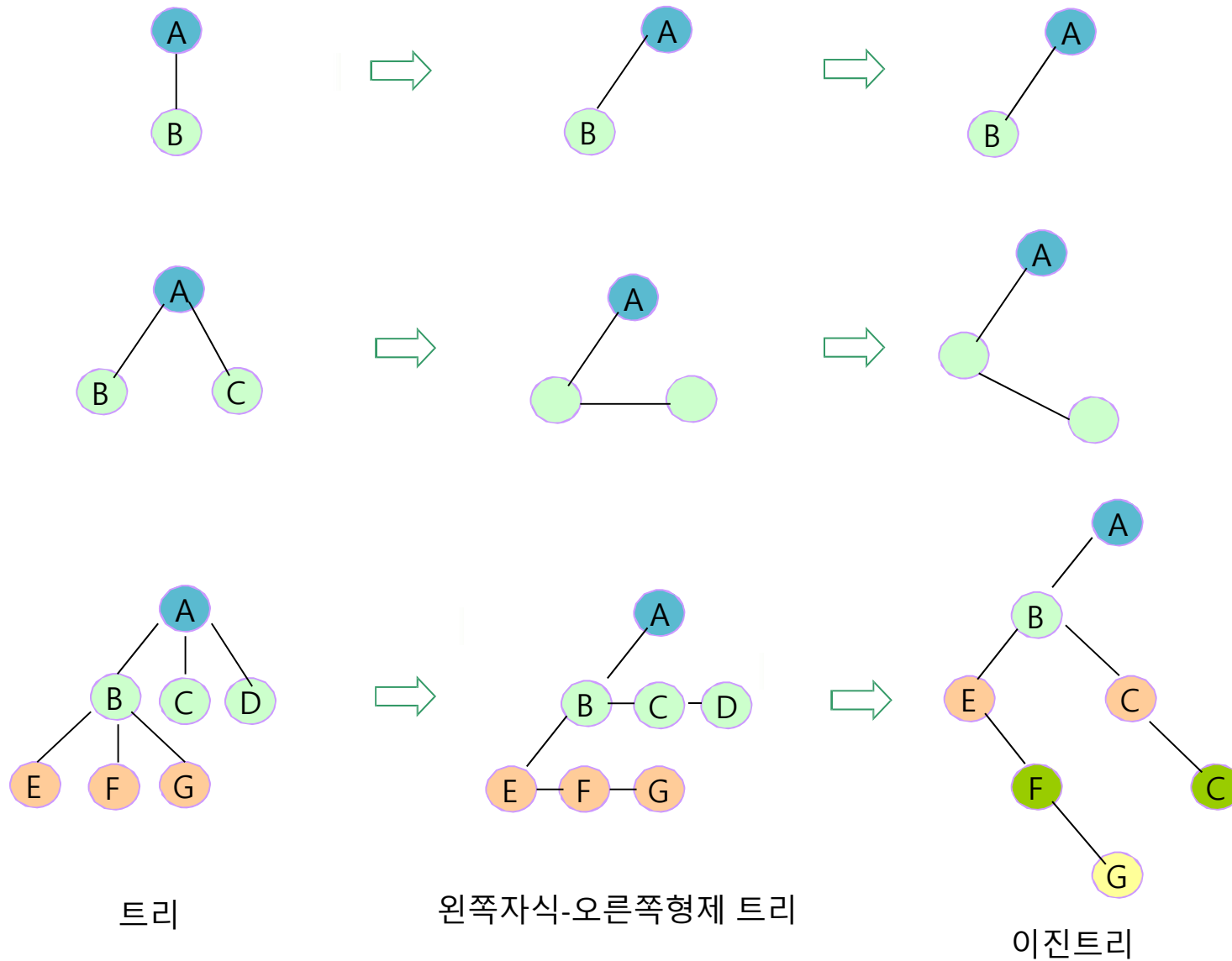
- 5) (1)과 (2)는 서로 다른 이진 트리

Contents

1. 트리의 이해

2. 이진 트리의 이해

트리와 이진 트리의 예



이진(Binary) 트리

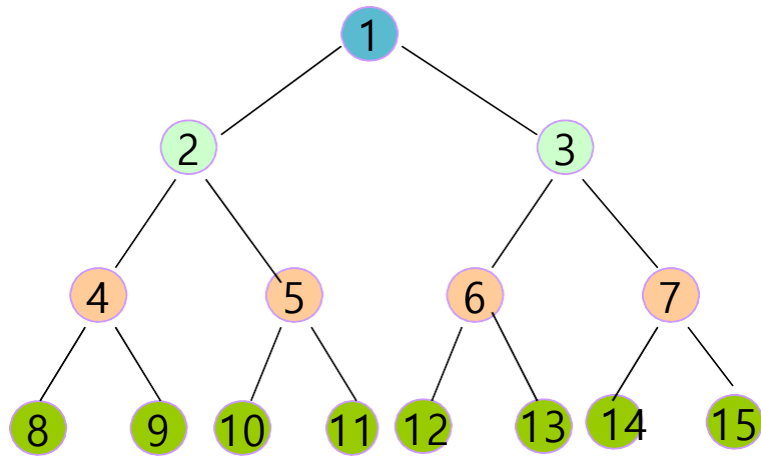
- 스큐(skewed) 이진 트리
 - 트리의 노드가 왼쪽이나 오른쪽으로 한쪽으로만 노드가 있는 트리
- 포화(full) 이진 트리(full binary tree of depth k)
 - 트리의 깊이가 k(레벨을 1로 시작), $2^k - 1$ 노드를 가진 이진 트리 (트리 깊이가 1이면 노드가 1개, 2이면 3개, 3이면 7개, 4이면 15개, ... 로 트리에 노드가 가득 찬 형태의 이진 트리)
- 완전(complete) 이진 트리(complete binary tree)
 - N개의 노드를 가진 complete(완전) 이진 트리는 포화 이진 트리에서 노드에 1부터 n까지 번호를 붙였을 때 만들어진 트리

Contents

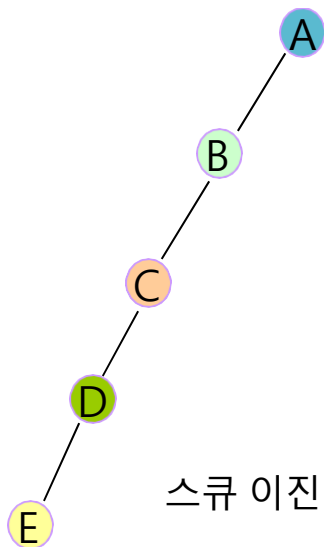
1. 트리의 이해

2. 이진 트리의 이해

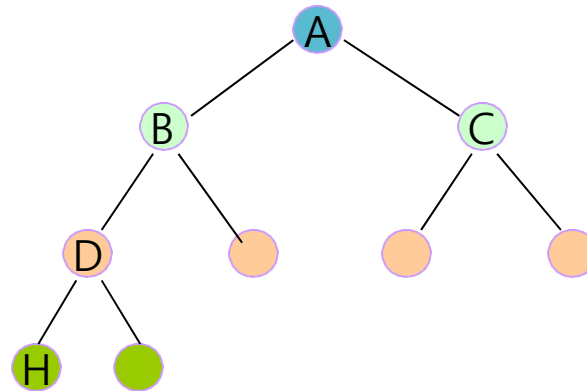
이진 트리의 종류



포화 이진 트리 (깊이 = 4)



스큐 이진 트리



완전 이진 트리

Q & A

