

## 8.1 트리의 기본 개념

교과목명	이산수학	분반		담당교수	김 외 현
학부(과)		학번		성명	

### 정의 트리

$A$ : 유한집합,  $T$ :  $A$ 에 대한 관계

$\forall v_0 \in A$ ,

(1)  $\exists_1$  경로:  $v_0 \rightarrow \forall v_i \in A - \{v_0\}$

(2)  $\nexists$  경로:  $v_0 \rightarrow v_0$  (cycle)

$\Leftrightarrow T$ : 트리

### 참고 내차수, 외차수

(1) 내차수

정점  $v$ 가 종점인 연결선의 개수

(2) 외차수

정점  $v$ 가 시점인 연결선의 개수

### 정리 트리의 성질

$(T, v_0)$ : 트리

(1)  $T$ : 어떤 사이클도 존재하지 않음

(2) 트리에서는 정점을 노드라고 함

(3)  $v_0$ : 유일한 루트 노드

(4)  $(v_0$ 의 내차수) = 0

(나머지 노드의 내차수) = 1

### 정의 서브(부분) 트리

$(T, v_0)$ : 루트 노드가  $v_0$ 인  $A$ 위에서의 트리

$v \in A$ ,  $T(v)$ : 루트 노드가  $v$ 인 트리

$\Rightarrow T(v)$ :  $(T, v_0)$ 의 서브 트리

### 정의 트리에서 사용되는 용어

(1) 루트 노드

내차수가 0인 노드

(2)  $\{v_i, v_j \in A \mid (v_i, v_j) \in T\}$

$\Rightarrow v_j$ :  $v_i$ 의 자식 노드

$v_i$ :  $v_j$ 의 부모 노드

형제 노드: 동일한 부모를 가지는 노드

(3) 잎(터미널) 노드

자식 노드가 하나도 없는 노드

(4) 조상 노드

루트 노드에서 그 노드에 이르는 경로에 나타난 모든 노드들

(5) 자손 노드

임의의 노드 하위에 연결된 모든 노드

(6) 레벨

(루트 노드의 레벨) = 0,

(어떤 노드의 레벨) =  $i$

$\Rightarrow$  (그 노드의 자식 노드의 레벨) =  $i + 1$

(7) 트리의 높이

트리에서 노드가 가질 수 있는 최대 레벨

(8) 차수

(노드의 차수) = (그 노드의 자식 노드의 수)

(트리의 차수) = (모든 노드 차수 중 최댓값)

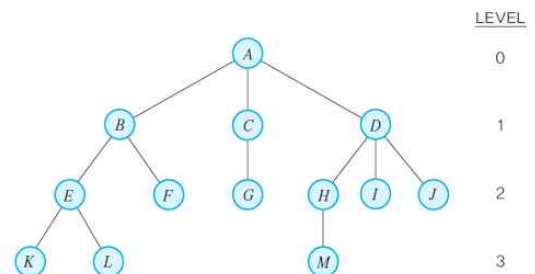
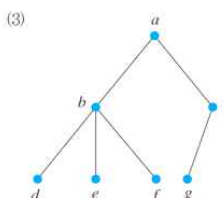
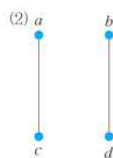
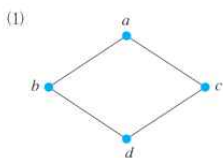
(9) 숲(포레스트)

서로 연결되지 않는 트리들의 집합으로서

트리에서 루트를 제거하면 숲을 얻을 수 있음

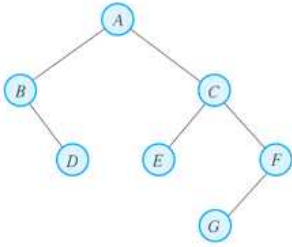
### 예제

1. 다음의 여러 그래프들이 트리에 해당되는지를 판단해보자.



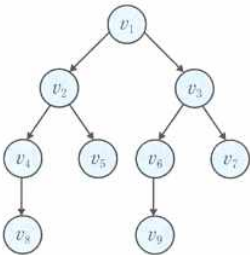
### 예제

2. 다음 트리에서 주어진 물음에 답해보자.



- (1) 잎 노드
- (2) 트리의 높이
- (3) C의 차수
- (4) G의 조상 노드

3. 다음 트리를 보고 노드  $v_1, v_3, v_4, v_6, v_7, v_8$ 의 레벨과 트리의 높이를 구해보자.



**정리**  $T$ :  $n$ 개의 노드를 가진 트리

$$\Rightarrow (T \text{ 연결선의 개수}) = n - 1$$

**정리**  $G = (V, E), |V| = n, |E| = m$

(1)  $G$ : 트리

$$\Leftrightarrow (2) \ G \text{는 연결되어있고 } m = n - 1$$

$$\Leftrightarrow (3) \ G \text{는 연결되어 있고 어느 한 연결선만 제거하더라도 } G \text{는 연결되지 않음}$$

$$\Leftrightarrow (4) \ G \text{는 사이클을 가지지 않으며 } m = n - 1$$

$$\Leftrightarrow (5) \ G \text{는 어느 한 연결선만 첨가하더라도 사이클을 형성}$$

**정의**  $n$ -트리

(1)  $T$ :  $n$ -트리

모든 노드가 최대  $n$ 개의 자식들을 가진 트리

(2)  $n = 2 \Rightarrow T$ : 이진 트리

(3) 완전  $n$ -트리

잎 노드를 제외한 모든 노드의 자식들이 정확히  $n$ 개인 트리

### 예제

4. 15개의 정점을 가지는 트리는 몇 개의 연결선을 가지는지 알아보자.

## 8.2 방향 트리

교과목명	이산수학	분반		담당교수	김 외 현
학부(과)		학번		성명	

### 정의 방향이 있고 순서화된 트리

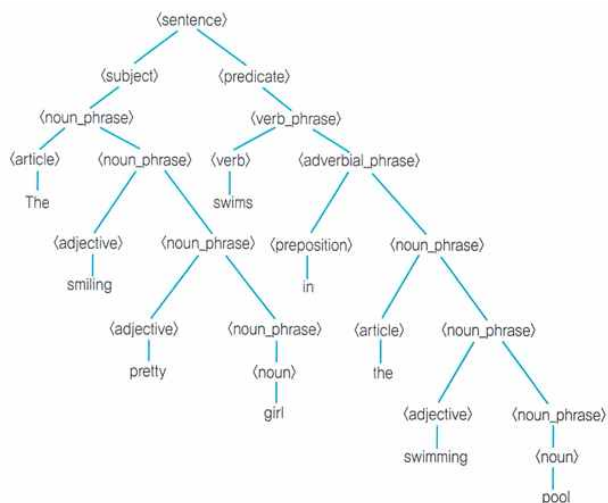
- (1) 선행자가 없는 루트라고 불리는 노드가 하나 있으나, 이 루트에서는 모든 노드로 갈 수 있는 경로가 있음
- (2) 루트를 제외한 모든 노드들은 오직 하나씩만의 선행자를 가짐
- (3) 각 노드의 후속자들은 통상 왼쪽으로부터 순서화됨

### 참고 방향 트리 그리는 법

- 어떤 방향 트리를 그릴 때 그 트리의 루트는 가장 위에 있음
- 아크(연결선)들은 밑을 향하여 그려짐
- 아크에 있는 화살표는 방향을 위해서는 필요하지 않으므로 화살표는 그리지 않음

### 예제 순서화된 트리 구조의 예

‘The smiling pretty girl swims in the swimming pool’에 대한 트리



## 8.3 이진 트리

교과목명	이산수학	분반		담당교수	김 외 현
학부(과)		학번		성명	

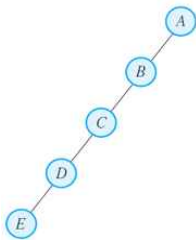
### 정의 이진 트리

각 노드의 최대 차수가 2인 트리

### 정의 이진 트리의 예

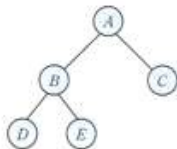
#### (1) 사향 이진 트리

왼쪽 또는 오른쪽으로 편향된 트리

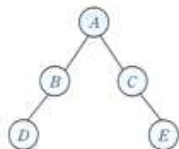


#### (2) 완전 이진 트리

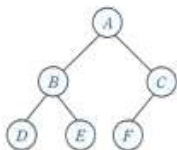
- (트리의 마지막 레벨을 제외한 모든 레벨의 노드의 차수) = 2
- (마지막 레벨의 노드 차수) = 2가 아니어도 됨
- 마지막 레벨의 노드가 왼쪽에서 오른쪽으로 차례대로 채워진 트리



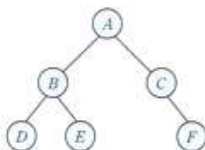
(a) 완전 이진 트리



(b) 완전 이진 트리 아님



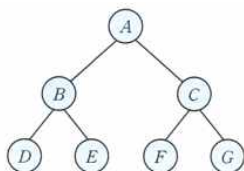
(c) 완전 이진 트리



(d) 완전 이진 트리 아님

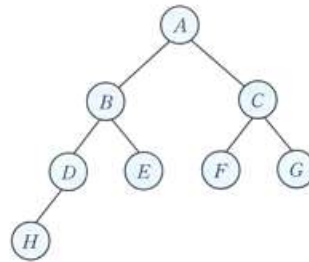
#### (3) 포화 이진 트리

잎 노드를 제외한 모든 노드가 2개의 자식 노드를 가지며 트리의 높이가 일정한 경우 즉, 완전이진트리에서 전체가 꽉 차있는 경우



### 예제

5. 다음 트리의 자식 노드가 몇 개인지 구하고, 루트 노드, 잎 노드, 트리의 차수를 구해보자. 그리고 각 노드의 레벨과 트리의 높이를 구하고, 이 트리가 이진 트리인지 확인해보자.



## 8.4 이진 트리의 표현

교과목명	이산수학	분반		담당교수	김 외 현
학부(과)		학번		성명	

### 정의 이진 트리를 표현하는 방법

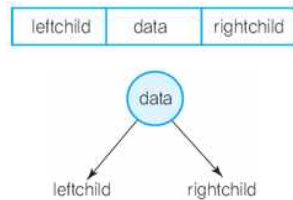
#### (1) 배열에 의한 방법

배열에 의한 방법은 트리의 중간에 새로운 노드를 삽입하거나 기존의 노드를 지울 경우 비효율적임

#### (2) 연결 리스트에 의한 방법

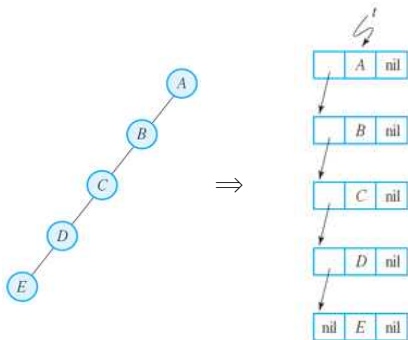
일반적으로 가장 많이 쓰이고 있는 방법

### 참고 노드의 구조

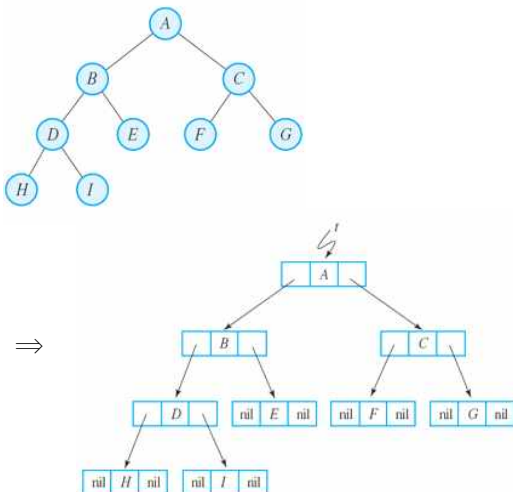


### 참고 이진 트리의 연결 리스트에 의한 표현

#### (1)



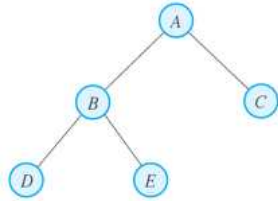
#### (2)



## 8.5 이진 트리의 탐방

교과목명	이산수학	분반		담당교수	김 의 현
학부(과)		학번		성명	

### 정의 트리 탐방 알고리즘



#### (1) 중순위 탐방

- ① 트리의 왼쪽 서브 트리를 탐방한다.
  - ② 트리의 루트 노드를 탐방한다.
  - ③ 트리의 오른쪽 서브 트리를 탐방한다.
- $\Rightarrow DBEAC$

#### (2) 전순위 탐방

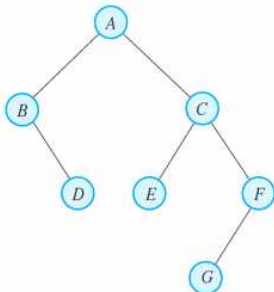
- ① 트리의 루트 노드를 탐방한다.
  - ② 트리의 왼쪽 서브 트리를 탐방한다.
  - ③ 트리의 오른쪽 서브 트리를 탐방한다.
- $\Rightarrow ABDEC$

#### (3) 후순위 탐방

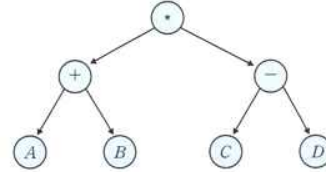
- ① 트리의 왼쪽 서브 트리를 탐방한다.
  - ② 트리의 오른쪽 서브 트리를 탐방한다.
  - ③ 트리의 루트 노드를 탐방한다.
- $\Rightarrow DEBCA$

### 예제

6. 다음의 이진 트리에서 중순위 탐방, 전순위 탐방, 후순위 탐방의 결과를 각각 구해보자.



### 정의 트리 표기법



#### (1) 중위 표기법

$$\Rightarrow A + B * C - D$$

#### (2) 전위 표기법

##### ① 일반 전위 표기법

$$\Rightarrow * (+ (A, B), - (C, D))$$

##### ② 케임브리지 폴란드식 표기법

$$\Rightarrow (* (+ AB) (- CD))$$

##### ③ 폴란드식 표기법

$$\Rightarrow * + AB - CD$$

#### (3) 후위 표기법(역폴란드식 표기법)

$$\Rightarrow AB + CD - *$$

### 예제

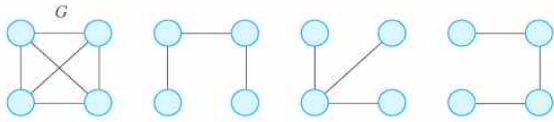
7. 대수식  $((a-b) * c - (d/e)) * f$ 를 트리표로 표현해보자. 그리고 트리에서 중순위 탐방, 전순위 탐방, 후순위 탐방의 결과를 각각 구해보자.

## 8.6 생성 트리와 최소 비용 생성 트리

교과목명	이산수학	분반		담당교수	김 외 현
학부(과)		학번		성명	

### 정의 생성 트리

그래프  $G$ 의 모든 노드들을 포함하는 트리



### 정의 생성 트리의 비용

트리 연결선의 값을 모두 합한 값

### 정의 최소 비용 생성 트리

최소한의 비용을 가지는 생성 트리

#### (1) 프림의 알고리즘

- ① 최소의 정점을 임의로 선택
- ② 정점의 간선들 중 최소 비용을 찾고, 해당 간선에 연결된 정점을 정점 집합에 추가
- ③ 이때 이미 지나온 정점과 연결되어 있거나 사이클이 발생하지 않는 간선을 선택
- ④ 정점 집합에서의 모든 정점에 연결된 간선들 중 최소비용을 찾아 반복

#### (2) 크루스칼의 알고리즘

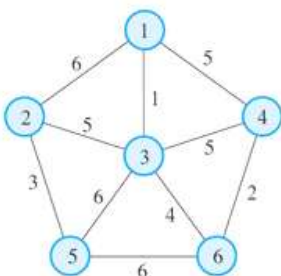
- ① 간선의 가중치를 오름차순으로 정렬한 집합을 생성
- ② 최소 가중치의 간선을 시작으로 정점을 순회
- ③ 사이클이 형성될 경우 해당 간선을 건너뛰

#### (1) 프림의 알고리즘

#### (2) 크루스칼의 알고리즘

### 예제

8. 다음 그래프에서 최소 비용 생성 트리를 구해보자.



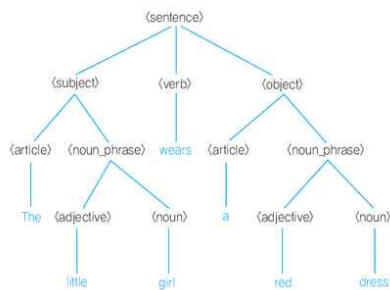
## 8.7 트리의 활용

교과목명	이산수학	분반		담당교수	김 의 현
학부(과)		학번		성명	

### 참고 트리의 응용 분야

#### (1) 문법의 파싱

- 트리는 문법의 파싱을 통하여 자연어 처리와 컴파일러 등에 활용됨
- 문장 트리에서 문장은 주어, 동사, 목적어로 이루어짐
- 주어는 관사와 명사구로 나누어지고 목적어는 관사와 명사구로 나누어짐
- 명사구는 형용사와 명사로 다시 나누어지는 파싱 과정을 반복함

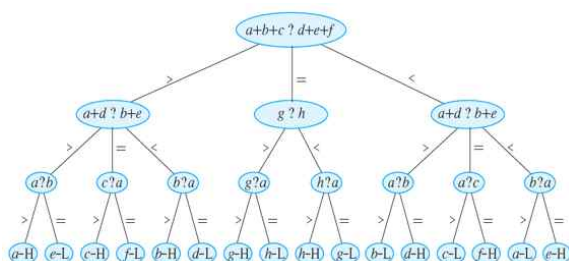


#### (2) 결정 트리

- 트리를 이용할 때 매우 유용하게 쓰임
- 가능성 있는 경우의 수가 너무나 많기 때문에 모든 면에서 입증하기가 매우 어려운 문제를 만날 수가 있음
- 결정 트리를 활용하면 주어진 문제를 일목요연하게 입증할 수 있음

### 예제 8개의 동전 문제

크기와 색깔이 똑같은 8개의 동전 a, b, c, d, e, f, g, h 중에서 한 개는 불량품이어서 다른 동전들과는 다른 무게를 가진다. 무게가 같거나 크고 작은 것만을 판단할 수 있는 하나의 천칭 저울을 사용하여 단 세 번만의 계량으로 어느 동전이 불량품이고, 다른 동전보다 무겁거나 가벼운지를 동시에 판단하고자 한다.



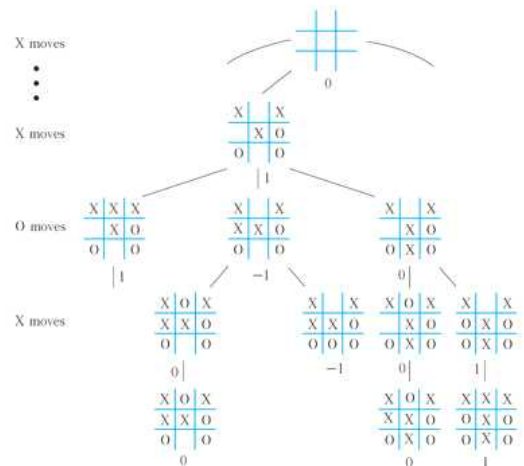
9. 크기와 모양이 똑같은 6개의 공이 있는데 이 중 5개의 무게는 같고 한 개는 약간 더 가볍다고 한다. 천칭 저울을 이용하여 최소 몇 번의 측정으로 1개의 가벼운 공을 찾아낼 수 있을까? 결정 트리를 만들어 찾아보자.

#### (3) 게임 트리

- 트리는 체스, 틱택토, 장기, 바둑 등 게임에 있어서의 진행과 전략을 구사할 수 있는 게임 트리로도 활용됨

### 예제 tic-tac-toe 게임

가로, 세로, 대각선으로 연속된 세 개를 놓으면 이기는 게임인 tic-tac-toe 게임의 일부





#### (4) 게임 트리의 인공지능 문제への 응용

- 다양한 가능성을 열어둔 채 폭넓게 사고하는 문제를 풀기 위해서는 트리를 이용하는 것이 매우 효율적임

#### 예제 8-puzzle 게임

3×3 크기의 박스 안에서 인접한 숫자를 빈 곳으로 계속적으로 움직여서 목표 상태로 만다는 서양식 게임

Start			Goal		
2	8	3	1	2	3
1	6	4	8	•	4
7	•	5	7	6	5

게임의 시작 상태에서 목적 상태로 가는 경우의 수를 나타낸 트리

