

# 최소 신장 트리 (Minimum Spanning Tree)

practice 자료구조/트리 · 2024. 10. 1. 19:03

Q. 그래프의 **최소 신장 트리 (Minimum Spanning Tree)** 알고리즘 중 **프림 (Prim) 알고리즘**과 **크루스칼 (Kruskal)** 알고리즘의 차이점과 사용 사례에 대해 설명해 주세요

A. 프림 알고리즘 (정점에 focus)

-임의의 정점(Node)에서 시작

-현재 트리에서 가장 가중치가 낮은 길을 따라 확장

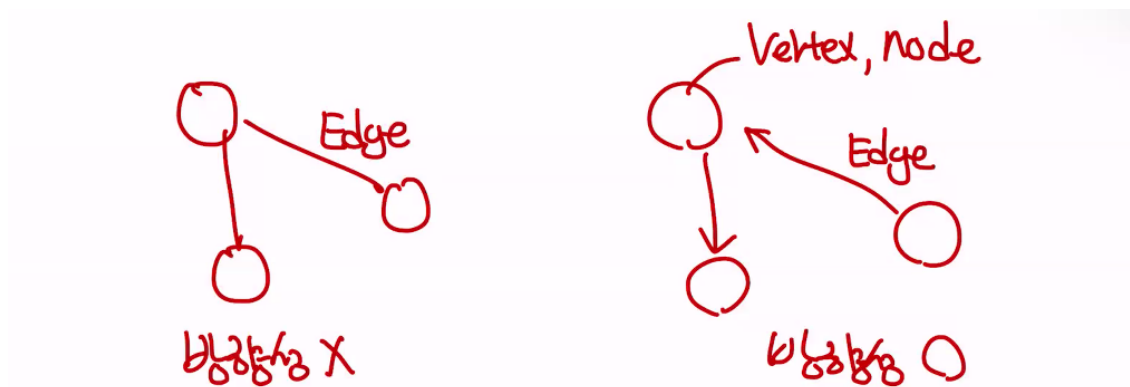
-모든 정점(노드)이 트리에 포함될때까지 반복 (또는 N-1간선)

A. 크루스칼 알고리즘 (간선 focus)

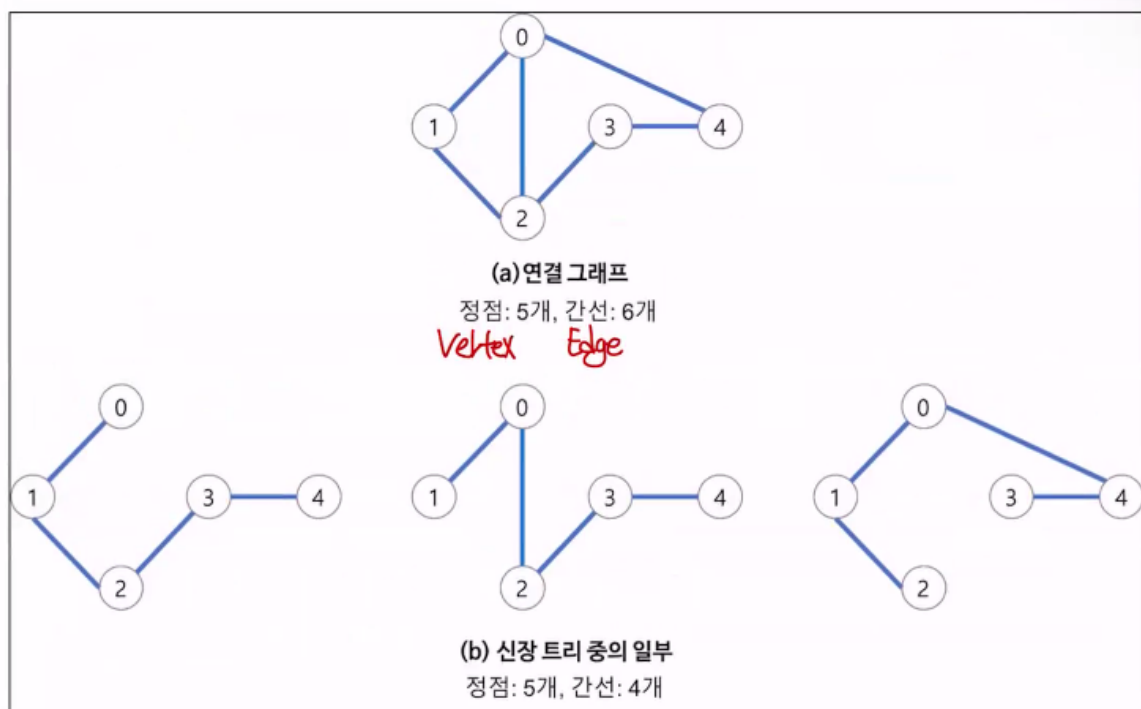
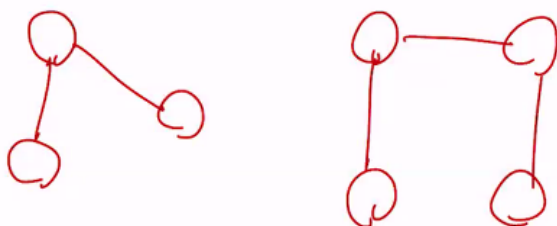
-가장 가중치 낮은 간선부터 순서대로 트리에 넣음

-사이클 발생시 하나 건너뛰다

-모든 정점이 트리에 포함될 때까지 반복



- 임의의 **Undirected 그래프**에, 모든 Vertex를 최소한의 Edge를 사용해 연결하는 것
- 이때 **Cycle이 있으면 안됨** (특정 지점에서 시작해서, 길을 따라가다 보니 시작 지점으로 돌아왔다 -> Cycle)
- 이때 **N개의 Vertex**가 있다면, 임의의 신장 트리(= 최소연결트리)는 **N - 1**개의 Edge를 사용했을 것

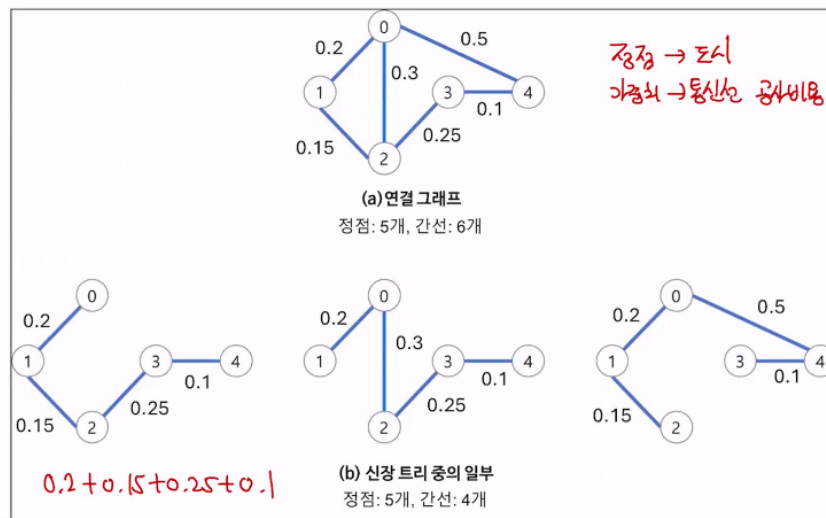


•DFS, BFS 등으로 모든 Vertex(정점,노드)를 탐색하고, 그 도중에 사용된 Edge(간선)만 모으면 만들 수 있음

•신장 트리는 통신망, 도로망 구축 등에 유용함

## 아래그림)

신장 트리들 중에서, 사용된 Edge들의 가중치 합이 최소인 트리를 최소 신장 트리라고 함



신장 트리들 중에서, 사용된 Edge들의 가중치 합이 최소인 트리를 최소 신장 트리라고 함

## 크루스칼 알고리즘이란?

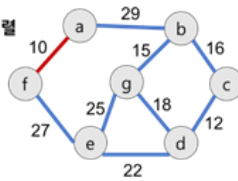
1. 주어진 그래프의 Edge들을 **가중치의 오름차순으로 정렬**한다 (**가중치가 낮은 Edge가 처음에 온다**)

2. 정렬된 Edge들에서, 순서대로 **Cycle**을 형성하지 않는 Edge들을 **선택**한다 (**신장 트리에 포함시킨다**)

3. 2번 동작을 모든 Vertex가 **신장 트리에 포함될 때까지 반복**한다

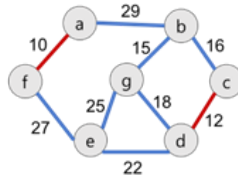
cf) 신장트리 -- 가중치가 없음 -> 최소신장트리: 가중치 합이 제일 작음 -> **크루스칼 알고리즘**

1) 간선들의 가중치 오름차순 정렬



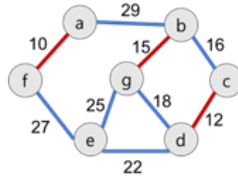
간선의 가중치	af	cd	bg	bc	dg	de	eg	ef	ab
	10	12	15	16	18	22	25	27	29

2)



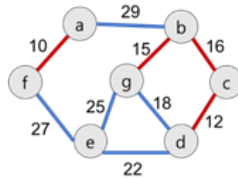
간선의 가중치	af	cd	bg	bc	dg	de	eg	ef	ab
	10	12	15	16	18	22	25	27	29

3)



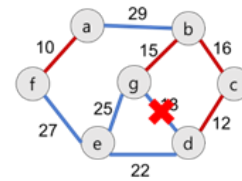
간선의 가중치	af	cd	bg	bc	dg	de	eg	ef	ab
	10	12	15	16	18	22	25	27	29

4)



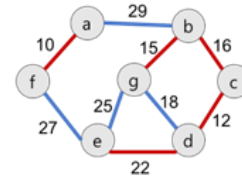
간선의 가중치	af	cd	bg	bc	dg	de	eg	ef	ab
	10	12	15	16	18	22	25	27	29

5) 사이클 형성. dg는 제외



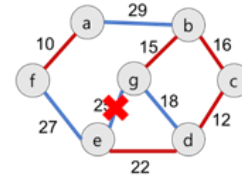
간선의 가중치	af	cd	bg	bc	dg	de	eg	ef	ab
	10	12	15	16	18	22	25	27	29

6)



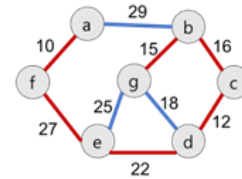
간선의 가중치	af	cd	bg	bc	dg	de	eg	ef	ab
	10	12	15	16	18	22	25	27	29

7) 사이클 형성. eg는 제외



간선의 가중치	af	cd	bg	bc	dg	de	eg	ef	ab
	10	12	15	16	18	22	25	27	29

8) N-1개의 간선 생성. 종료



간선의 가중치	af	cd	bg	bc	dg	de	eg	ef	ab
	10	12	15	16	18	22	25	27	29

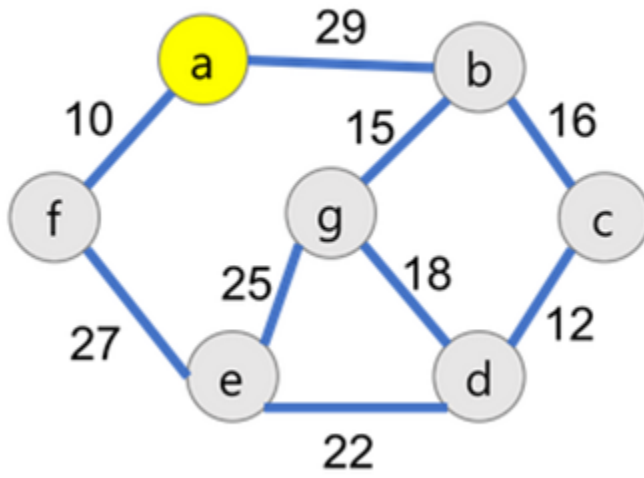
## 프림 알고리즘이란?

1. 시작 Vertex를 신장 트리에 포함시킨다

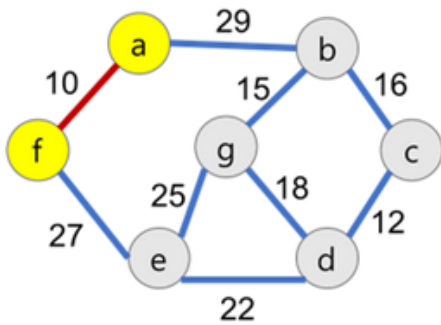
2. 현재 신장 트리에 포함된 Vertex들에 인접한 Vertex들 중에서, 가장 낮은 가중치로 연결된 Vertex를 신장 트리에 포함시킨다

3. 2번 작업을 모든 Vertex가 신장 트리에 포함될 때까지 반복한다

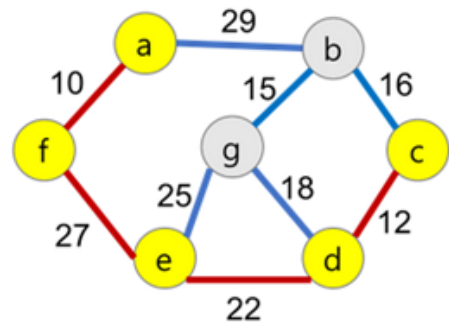
### 1) 시작 단계: 시작 정점만 포함



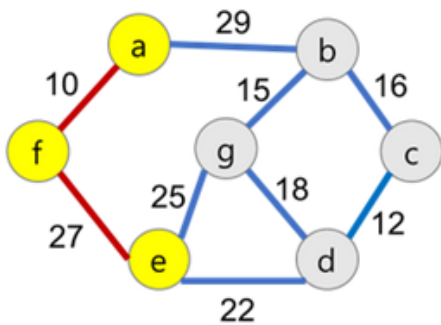
2) 인접 정점 중 최소 간선으로 연결된 정점 선택



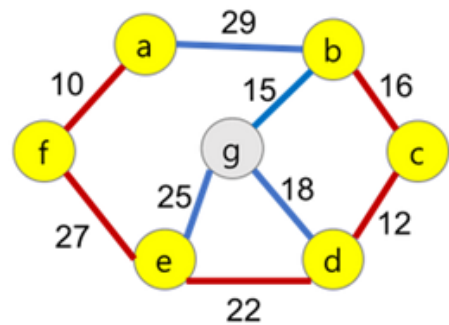
5)



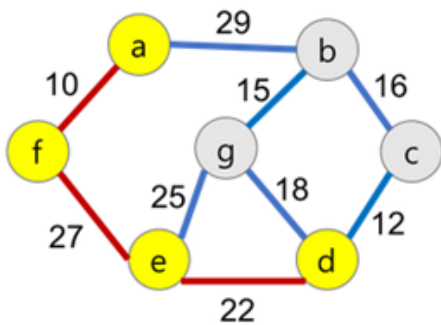
3)



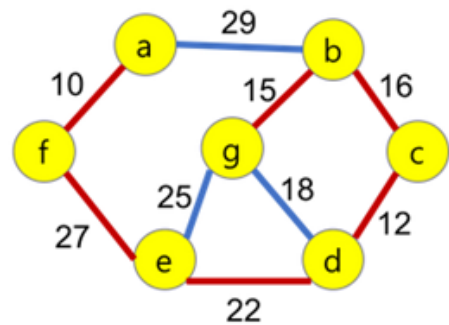
6)



4)



7) N-1개의 간선 생성. 종료



크루스칼 알고리즘 - 간선의 영향받음

•크루스칼 알고리즘은 먼저 모든 Edge를 가중치 순으로 정렬해야 한다 (시간 복잡도에 가장 큰 영향)

•Edge의 수를 E라 할때, 효율적인 정렬 알고리즘을 사용하면, 정렬의 시간 복잡도는  $O(E \log E)$ 가 된다

## 프림 알고리즘 - 노드영향받음

- 프림 알고리즘에서는 (Vertex의 수를  $V$ 라 할때)

- 1.하나씩 모든 Vertex를 신장 트리에 포함시킴 (총  $V$ 번 반복)

- 2.각 단계마다 모든 Vertex를 검색해서 가장 낮은 가중치로 연결된 Vertex 찾기 (총  $V$ 번 반복)

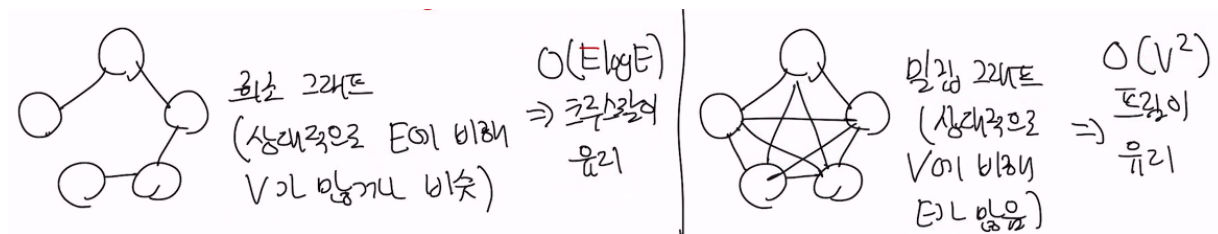
- 따라서 시간 복잡도는  $O(V^2)$ 가 된다

- 크루스칼 알고리즘의 시간 복잡도 =  $O(E \log E)$

- 프림 알고리즘의 복잡도 =  $O(V^2)$ 가 된다

- 만약  $V$ 가 크다면 (희소 그래프) 크루스칼이 유리

- $V$ 작으면 (밀집 그래프) 프림 알고리즘이 유리

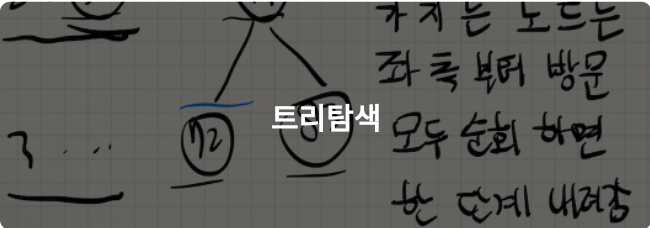
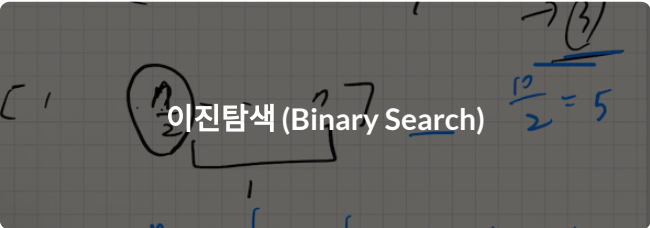
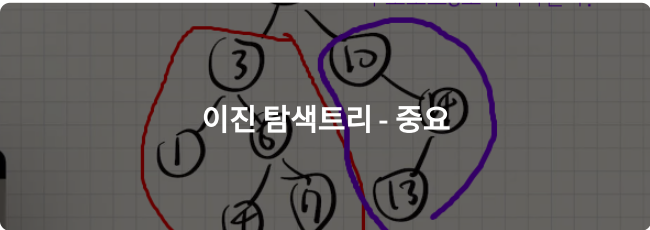


♡ 공감 📌 🗨

구독하기

'practice' 자료구조 > 트리 카테고리의 다른 글

<a href="#">AVR 트리</a> (0)	2024.09.22
<a href="#">이진 탐색트리 - 중요</a> (0)	2024.09.22
<a href="#">이진탐색 (Binary Search)</a> (0)	2024.09.22
<a href="#">트리탐색</a> (0)	2024.09.22
<a href="#">그래프, 트리</a> (0)	2024.09.20



## 자연어(NLP)

네이쳐2024 님의 블로그입니다.

구독하기 +



댓글 0



이름

비밀번호

내용을 입력하세요.



등록