

---

“본 강의 동영상 및 자료는 대한민국 저작권법을 준수합니다. 본 강의 동영상 및 자료는 상명대학교 재학생들의 수업목적으로 제작·배포되는 것이므로, 수업목적으로 내려받은 강의 동영상 및 자료는 수업목적 이외에 다른 용도로 사용할 수 없으며, 다른 장소 및 타인에게 복제, 전송하여 공유할 수 없습니다. 이를 위반해서 발생하는 모든 법적 책임은 행위 주체인 본인에게 있습니다.”

---

# 5. Greedy algorithm

---

## 5.0 Basics

## 5.1 Minimum spanning trees

## 5.2 Knapsack problem

## 5.3 Job sequencing with deadline

## 5.4 Optimal merge patterns

## 5.5 Huffman encoding

# 5.1 Minimum cost spanning tree

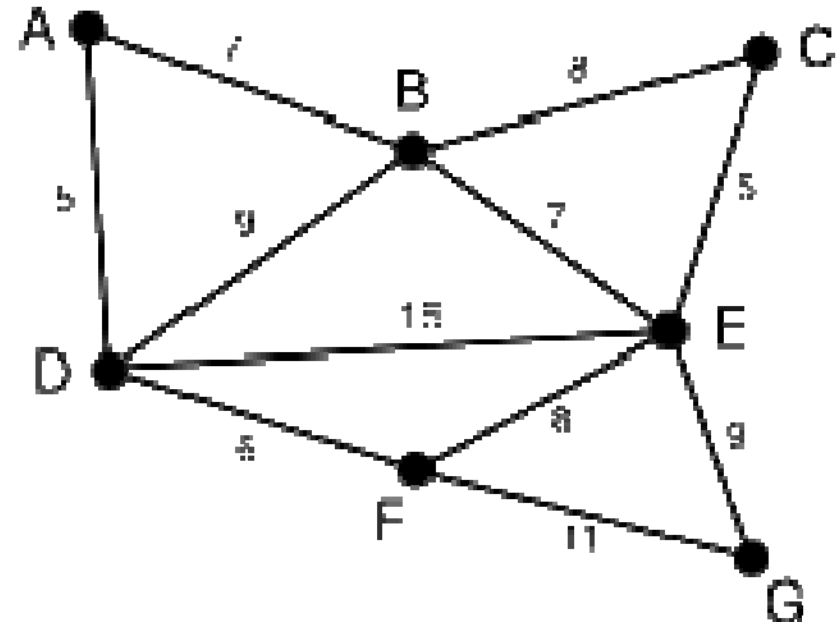
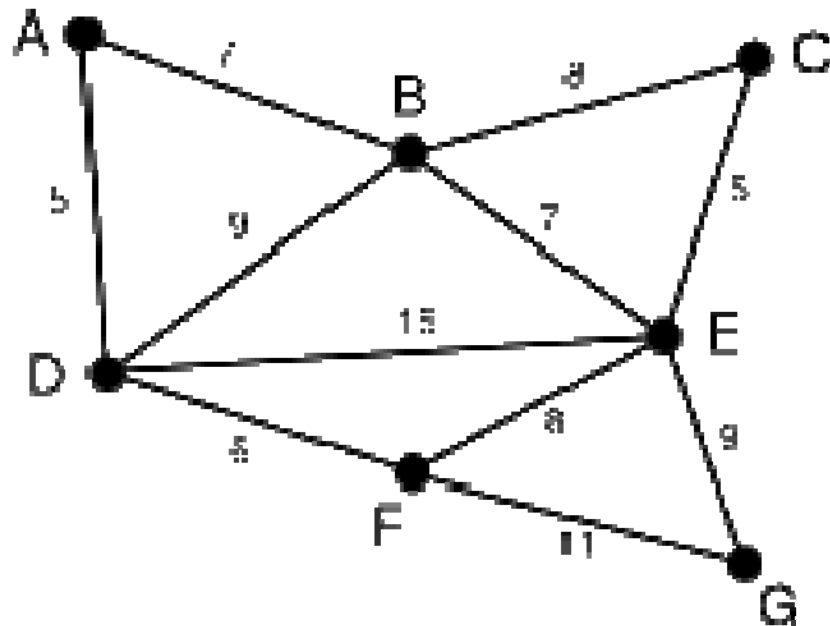
---

- Spanning tree
  - Connecting every vertex of a graph  $G$  using minimum number of edges
  - If  $|V| = n$ , then at least  $(n-1)$  edges are required
  - Does not allow a cycle
  - Spanning trees through graph search algorithms
    - Depth-first search  $\rightarrow$  Depth-first spanning tree
    - Breadth-first search  $\rightarrow$  Breadth-first spanning tree

## 5.1 Minimum cost spanning tree

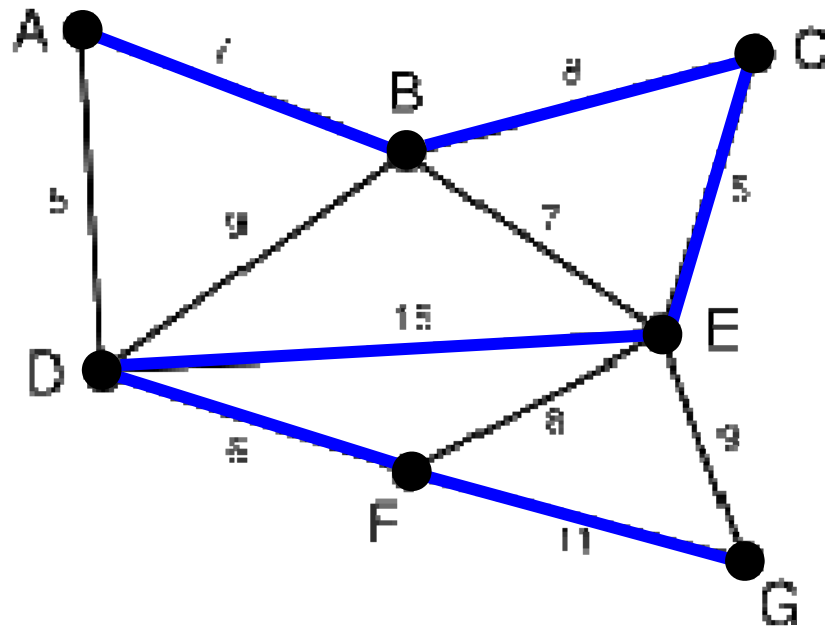
---

- Ex)

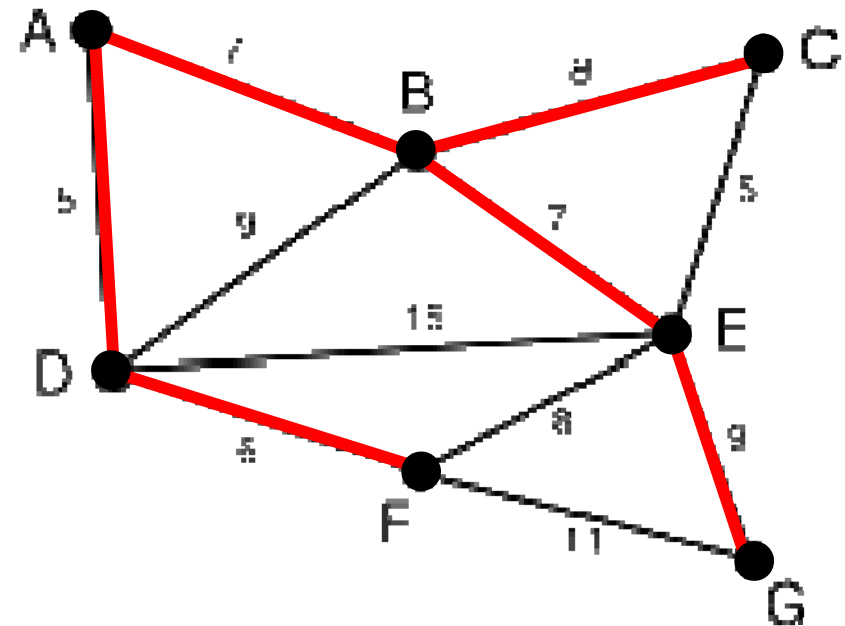


# 5.1 Minimum cost spanning tree

- Ex) Spanning trees



Depth-first spanning tree



Breadth-first spanning tree

# 5.1 Minimum cost spanning tree

---

- Minimum cost spanning tree
  - Computed on a weighted graph
  - A spanning tree whose sum of edge weights is minimum.
  - Properties
    - No new edges.
    - $(n-1)$  edges for  $|V| = n$ .
    - No cycle.

# 5.1 Minimum cost spanning tree

---

Problem:  
Connect  
all the  
vertices

Solution

# 5.1 Minimum cost spanning tree

---

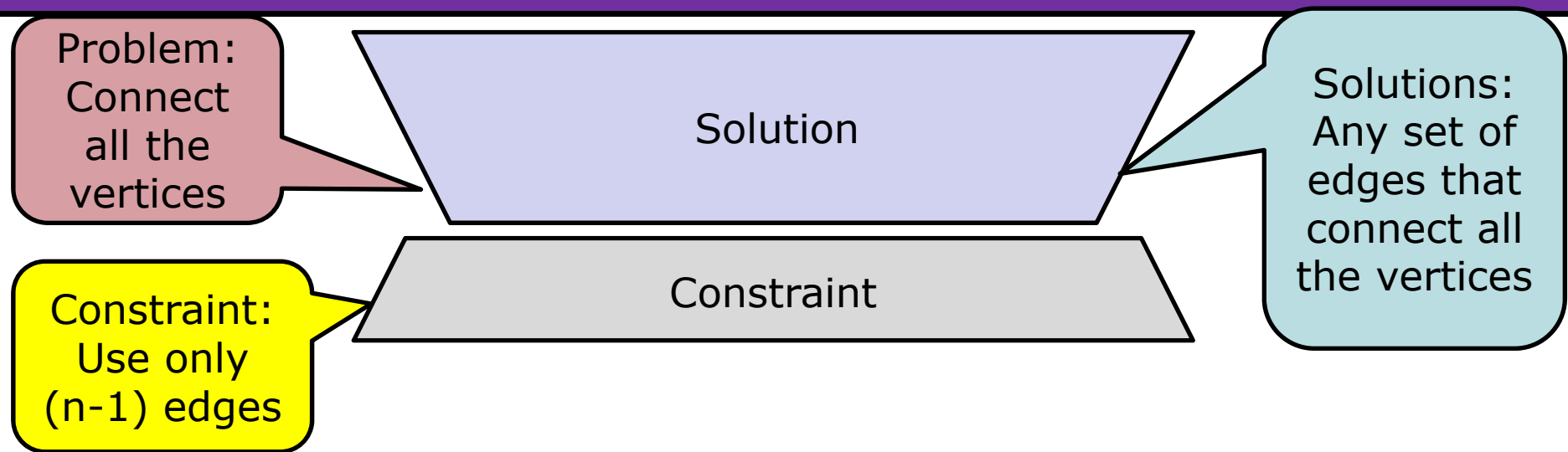
Problem:  
Connect  
all the  
vertices

Solution

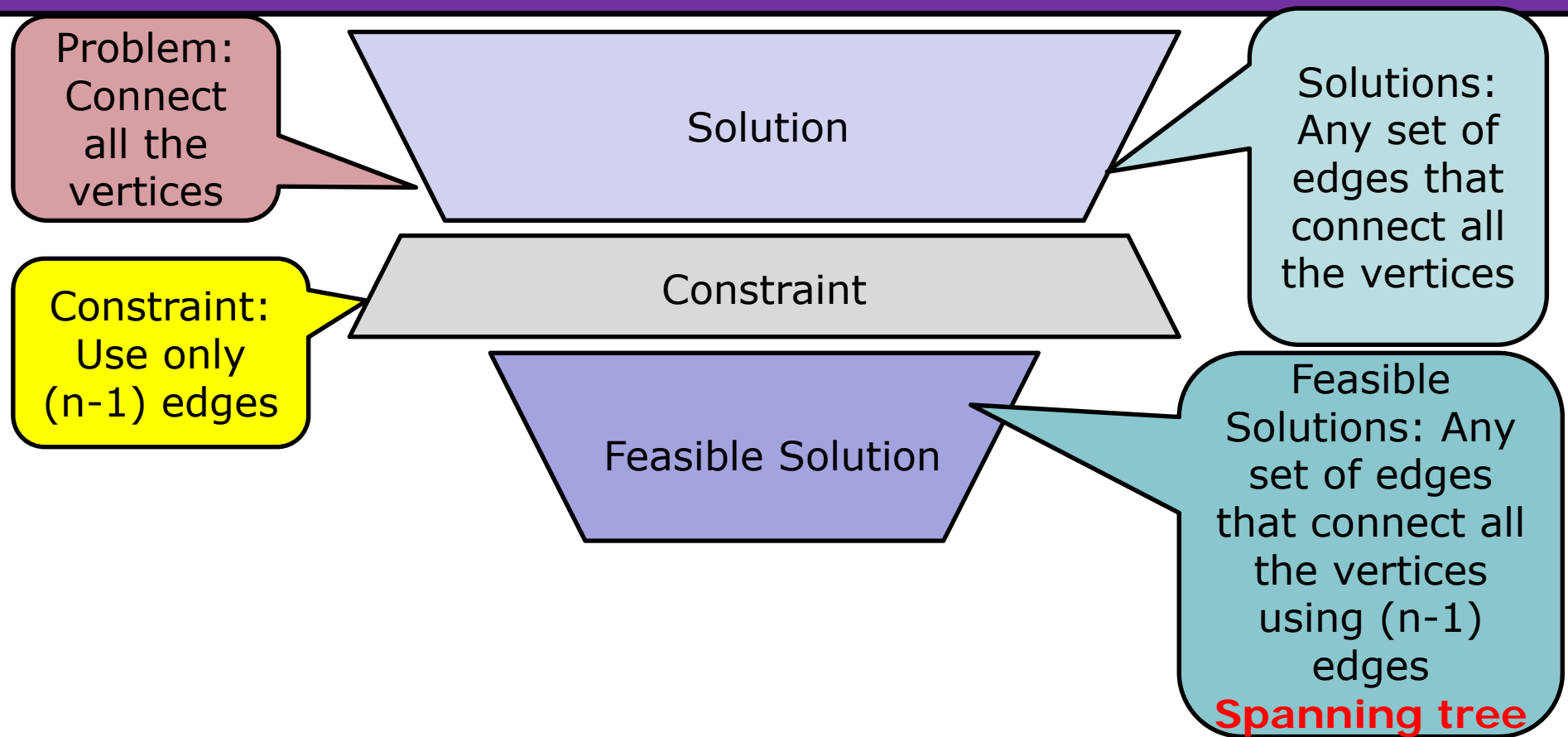
Solutions:  
Any set of  
edges that  
connect all  
the vertices



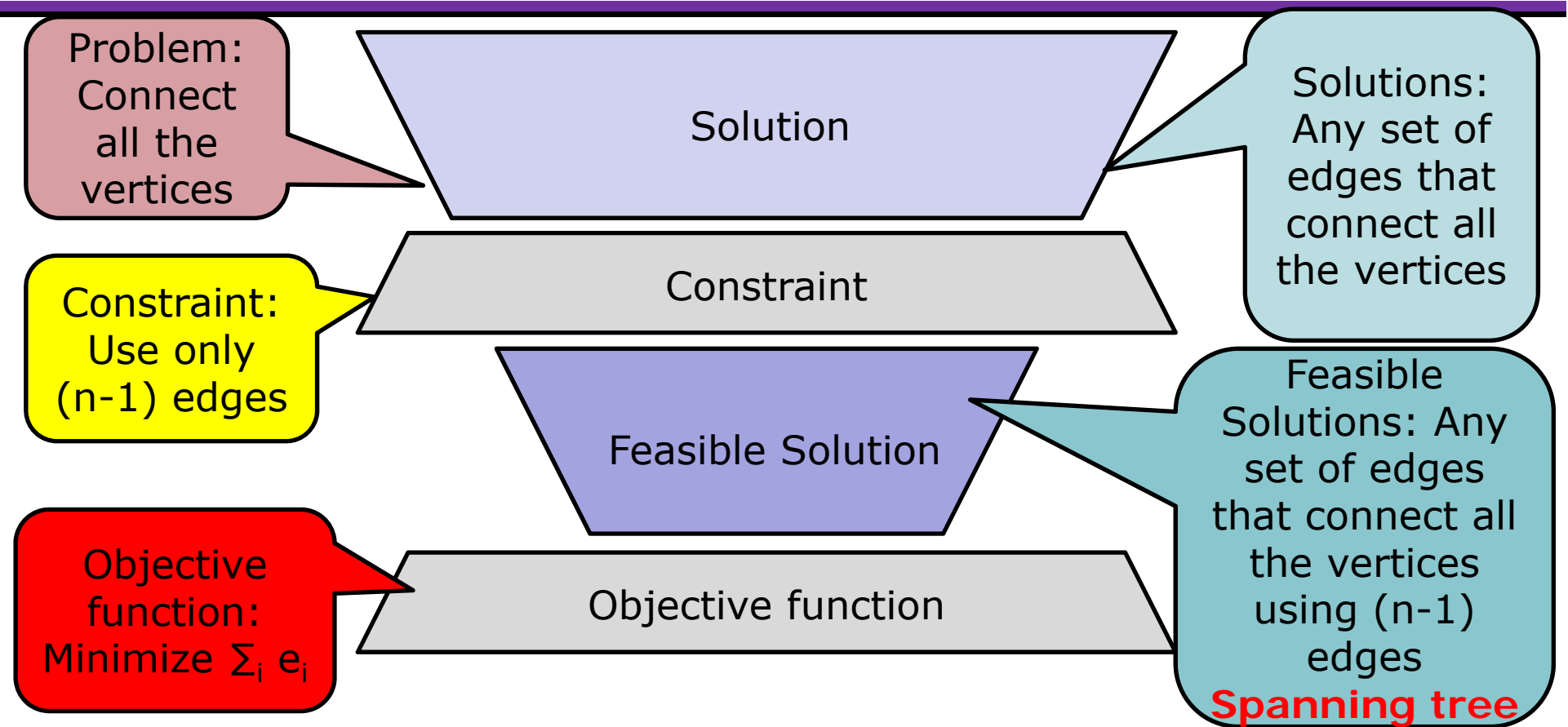
# 5.1 Minimum cost spanning tree



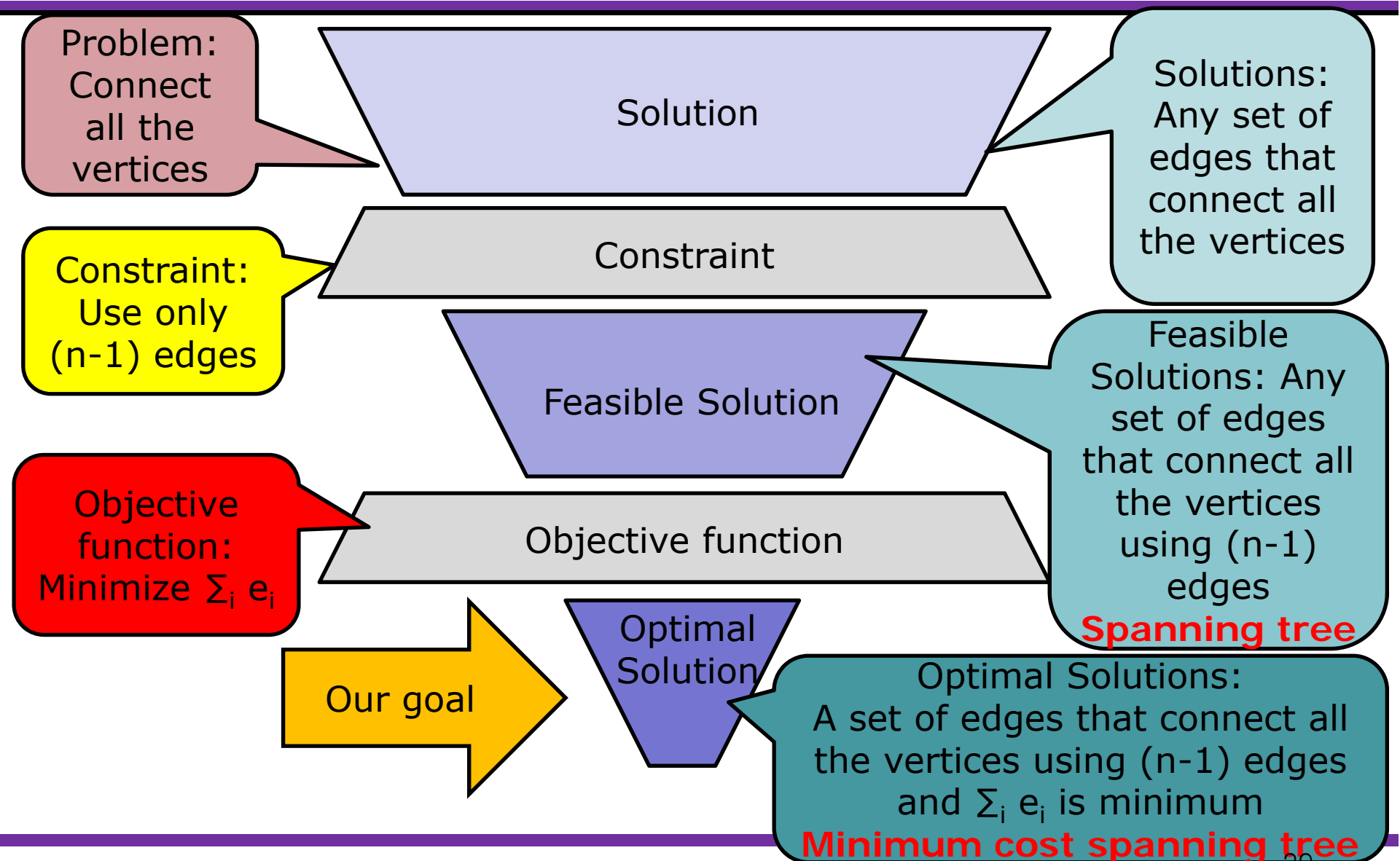
# 5.1 Minimum cost spanning tree



# 5.1 Minimum cost spanning tree



# 5.1 Minimum cost spanning tree



## 5.1 Minimum cost spanning tree

---

다음 설명 중 옳은 것을 모두 고르시오.

- (a)  $G = (V, E)$ 인 그래프의 spanning tree  $T$ 를  $(V', E')$ 라고 표현하면  $V = V'$ 이고  $E' \subseteq E$ 이다.
- (b) vertex의 수가  $n$ 개인 graph의 minimum cost spanning tree의 edge의 수는  $n-1$ 개이다.
- (c) Depth-first spanning tree는 depth-first search를 이용해서 얻어지는 minimum cost spanning tree이다.
- (d) Minimum cost spanning tree를 구하는 알고리즘은 breadth-first spanning tree를 구하는 알고리즘보다 더 많은 계산 시간을 요구한다.