5. Greedy algoritm

5.0 Basics

5.1 Minimum spanning trees

5.2 Knapsack problem

5.3 Job sequencing with deadline

5.4 Optimal merge patterns

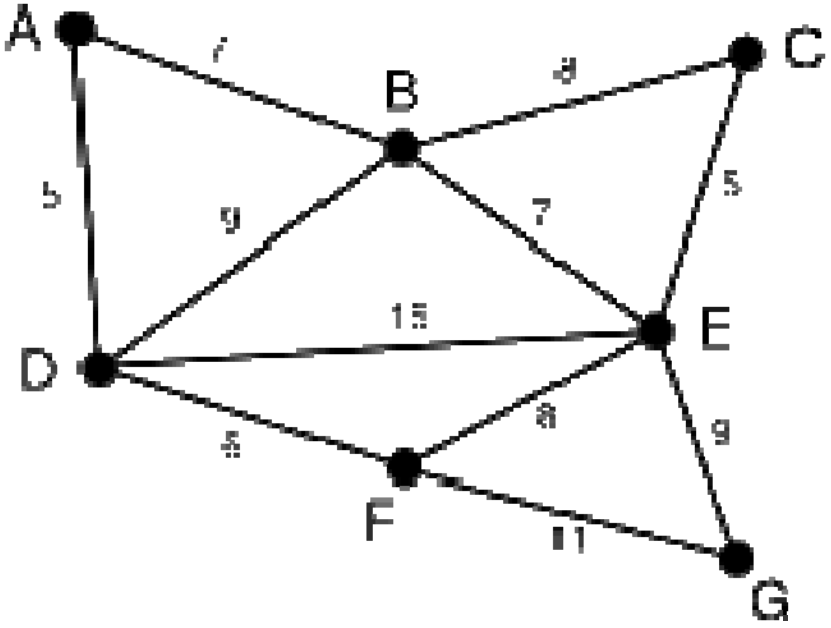
5.5 Huffman encoding

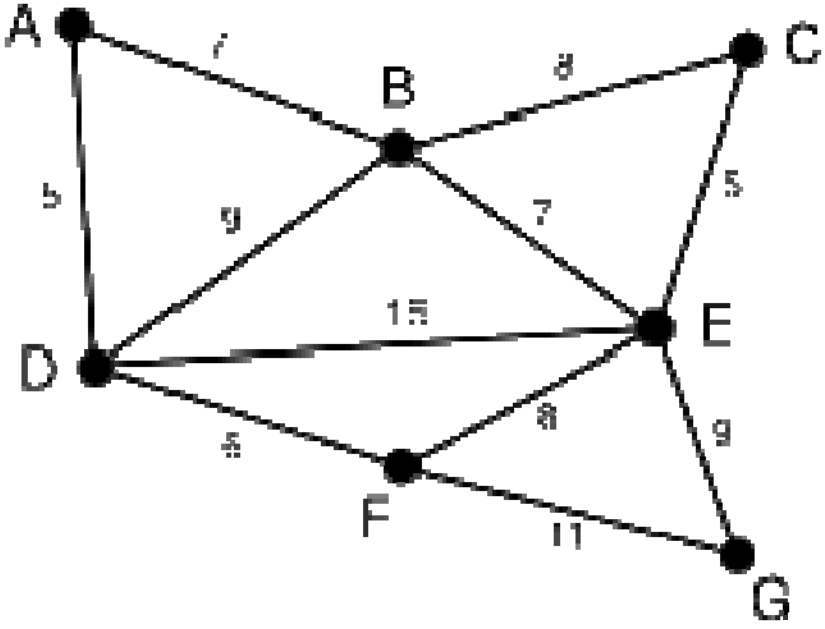
5.1 Minimum cost spanning tree

- 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 → Depth-first spanning tree
    - Breadth-first search → Breadth-first spanning tree

5.1 Minimum cost spanning tree

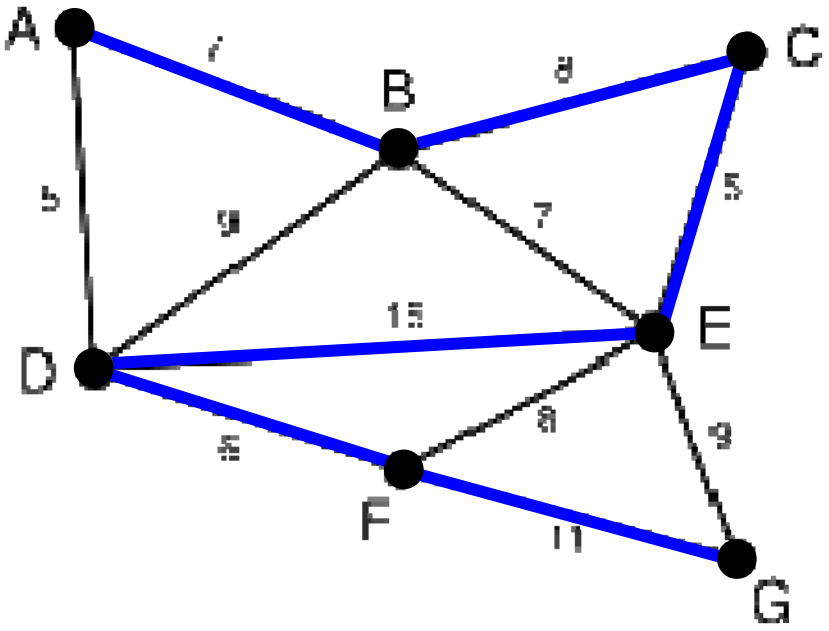
- Ex)



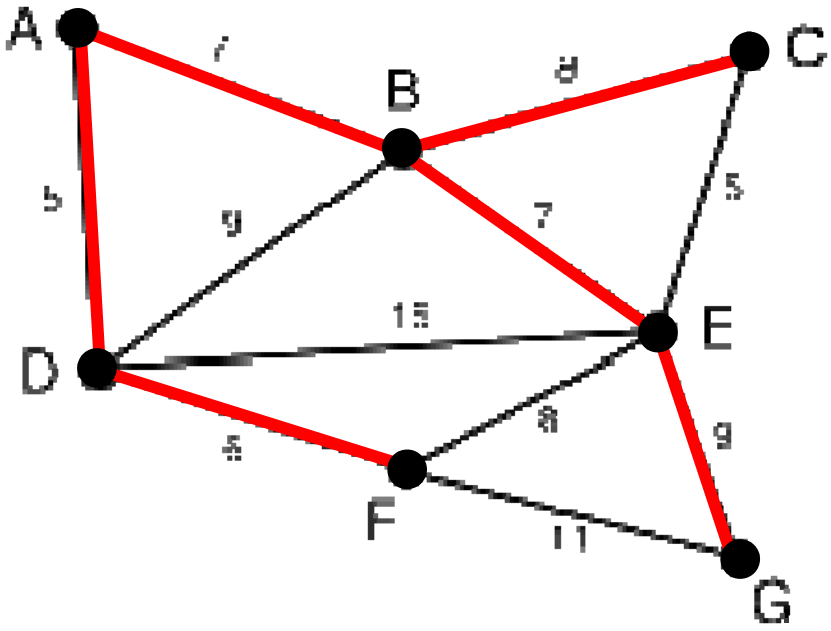


5.1 Minimum cost spanning tree

- Ex) Spanning trees



Depth-first spanning tree

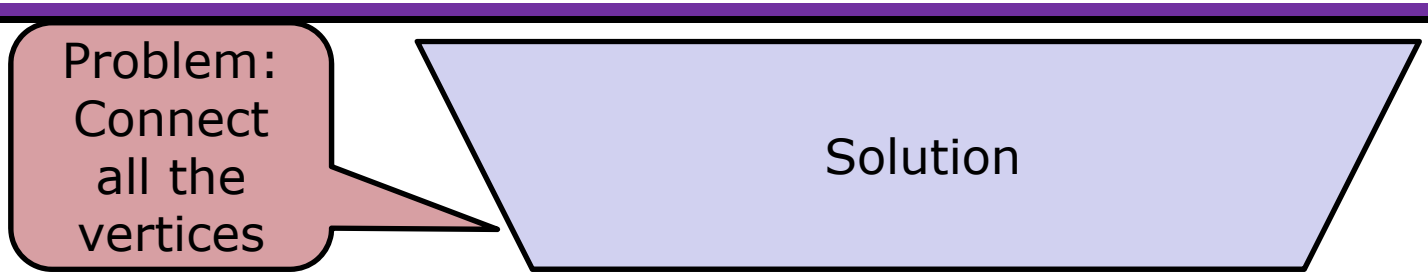


Breadth-first spanning tree

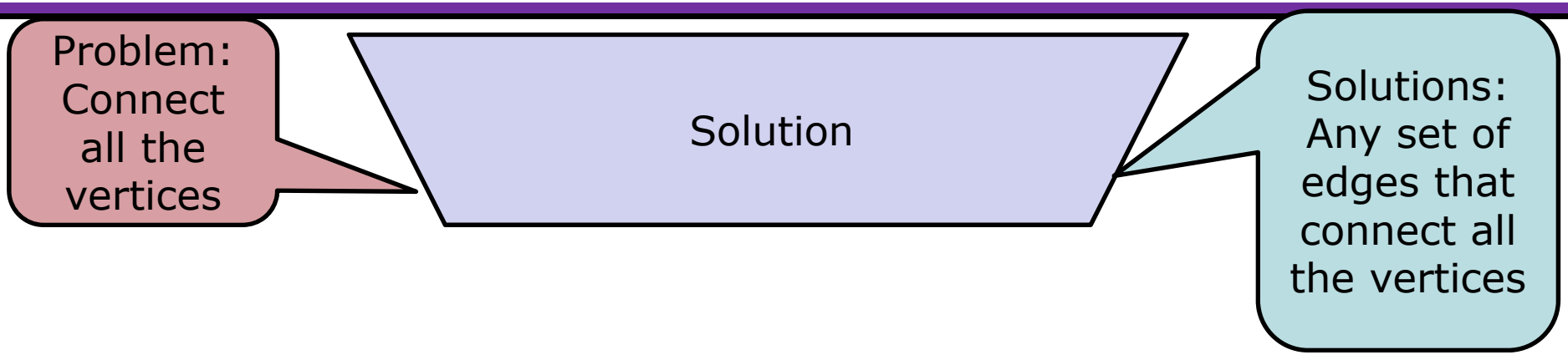
5.1 Minimum cost spanning tree

- 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.

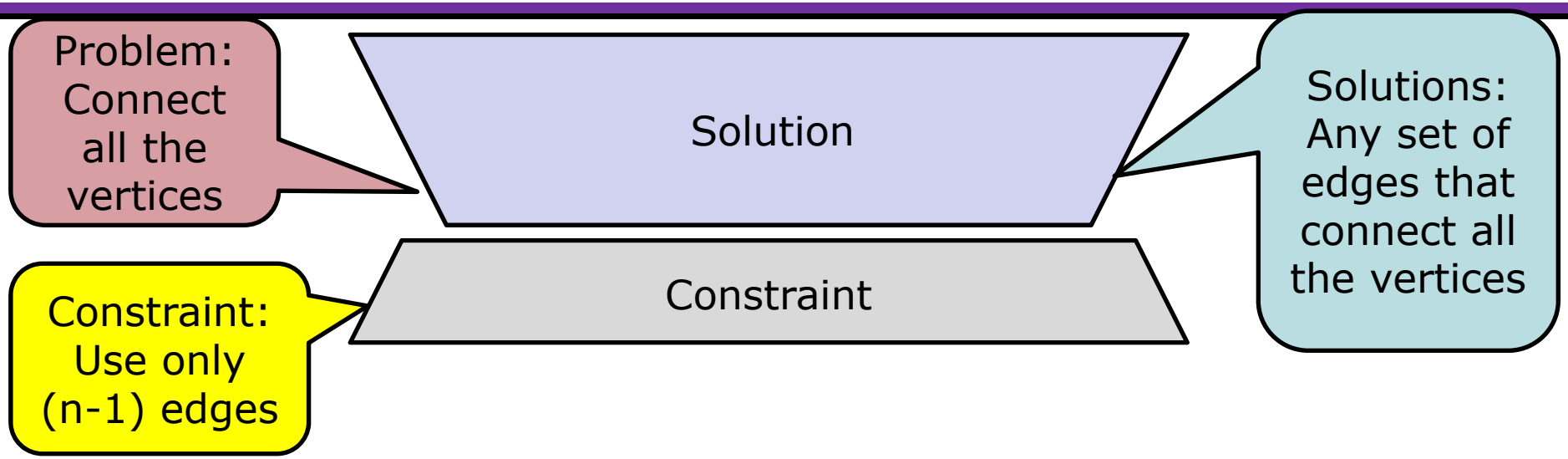
5.1 Minimum cost spanning tree



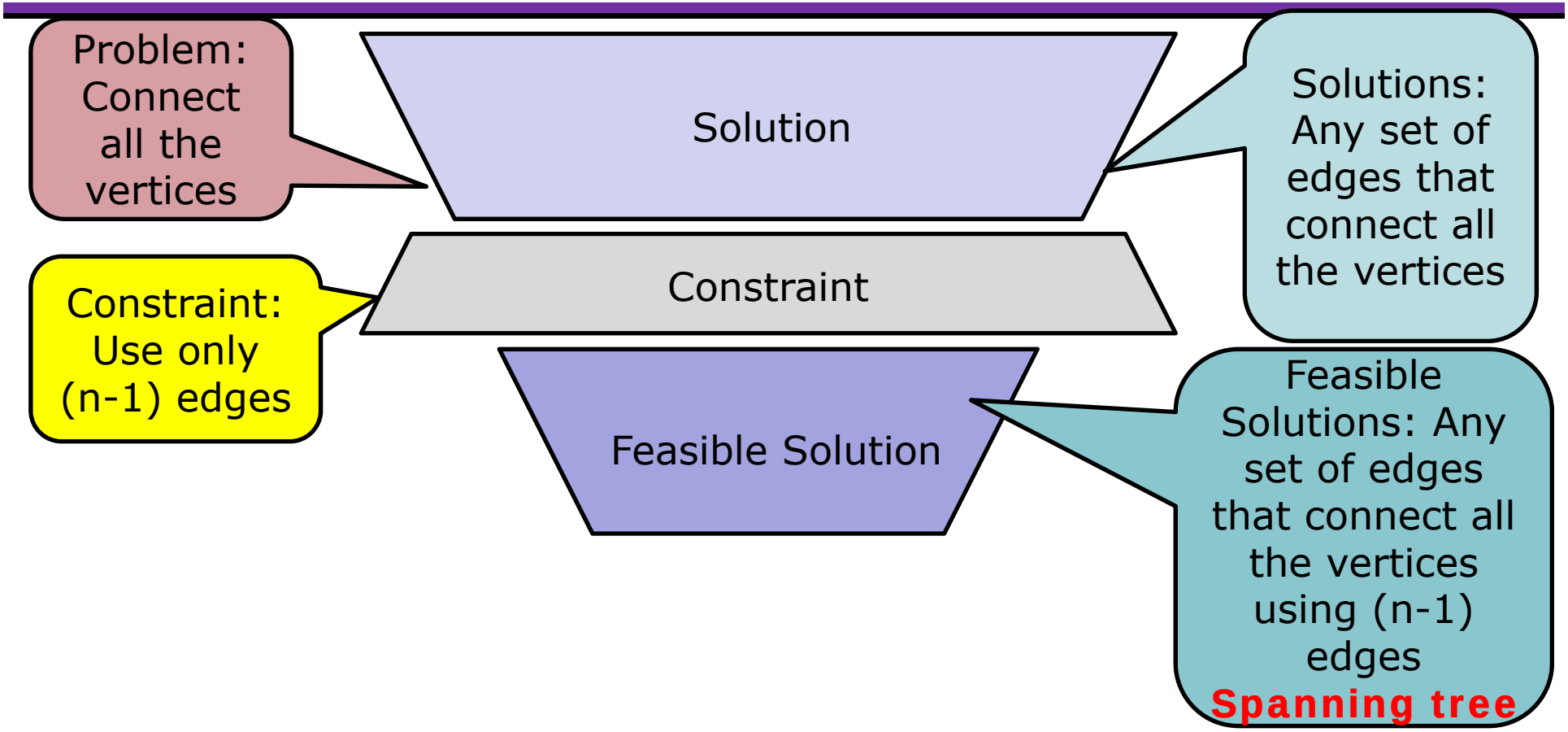
5.1 Minimum cost spanning tree



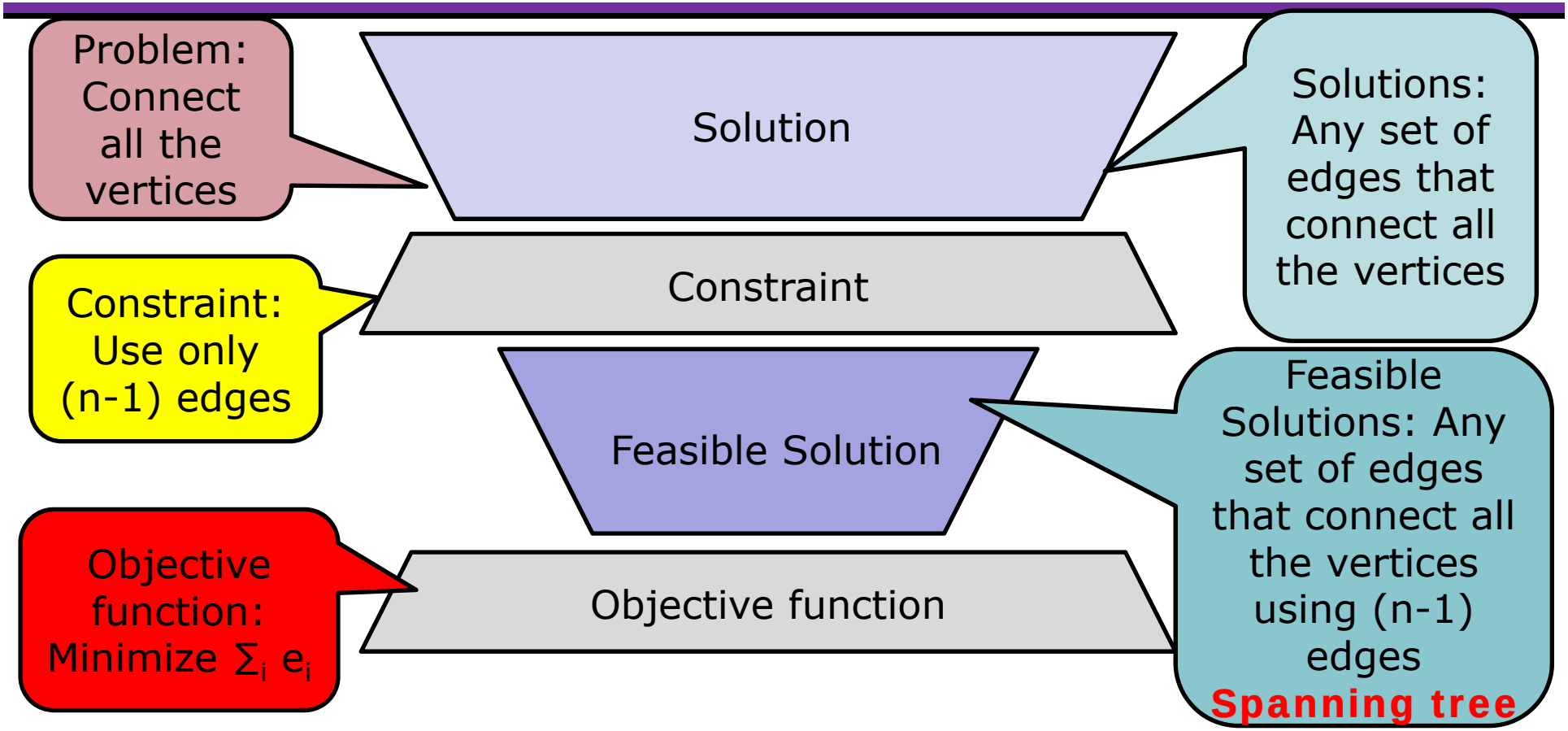
5.1 Minimum cost spanning tree



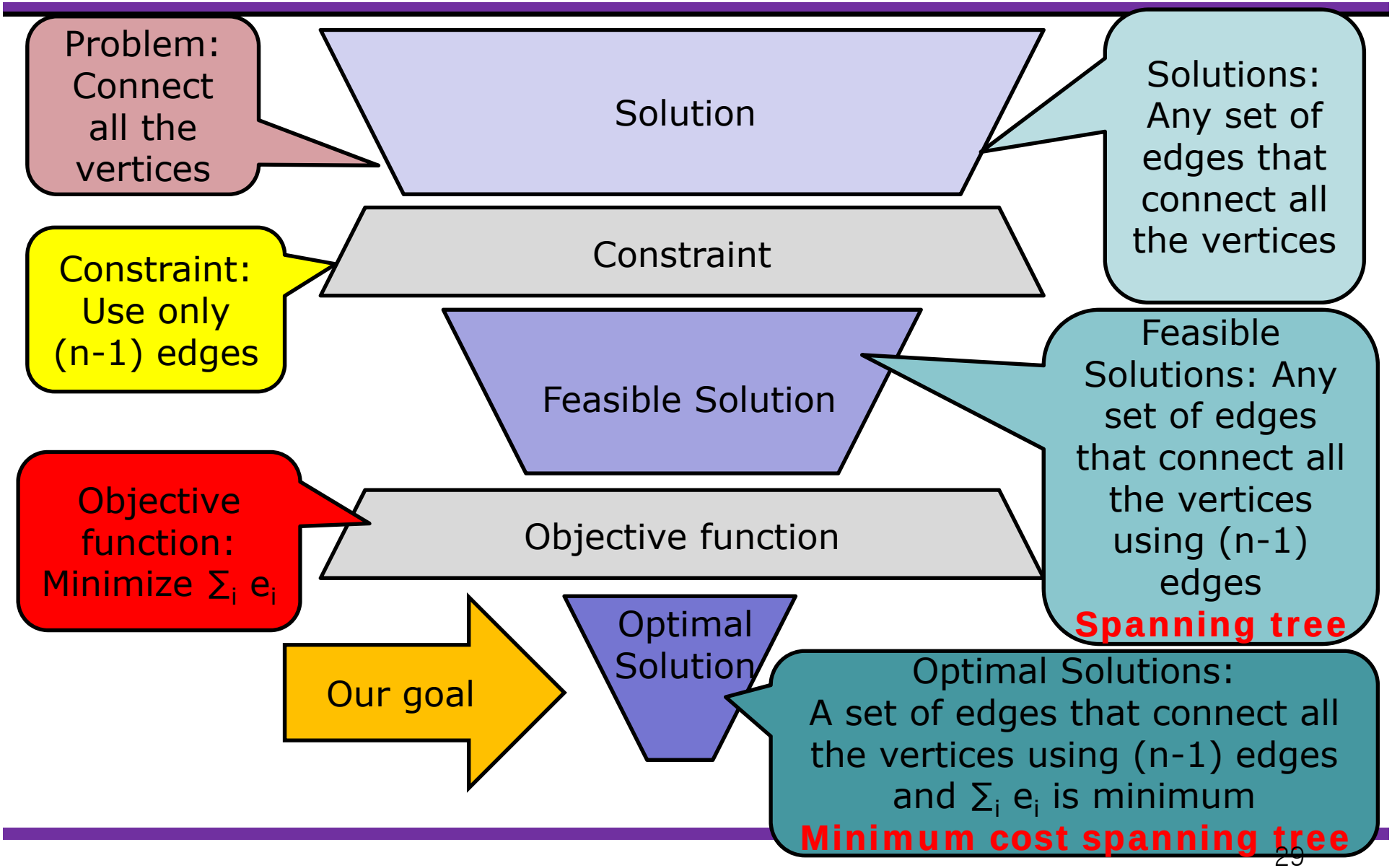
5.1 Minimum cost spanning tree



5.1 Minimum cost spanning tree



5.1 Minimum cost spanning tree



5.1 Minimum cost spanning tree

다음 설명 중 옳은 것을 모두 고르시오.

(a) G = (V, E)인 그래프의 spanning tree T를 (V', E')라고 표현하면 V = V'이고 E' ⊆ E이다.

(b) vertex의 수가 n개인 graph의 minimum cost spanning tree의 edge의 수는 n-1개이다.

(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.