알고리즘

4. Graph

상명대학교 컴퓨터과학과

민 경 하

Contents

4. Graph

4.0 Introduction

4.1 Why graph?

4.2 Depth-first search in undirected graph

4.3 Depth-first search in directed graphs

4.4 Strongly connected components

4.5 Biconnected component

4.6 Distances

4.7 Breadth-first search

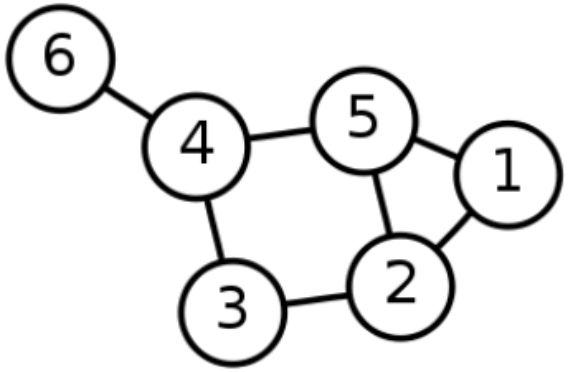
4.8 Single source shortest path

4.9 All pairs shortest path

4.0 Introduction

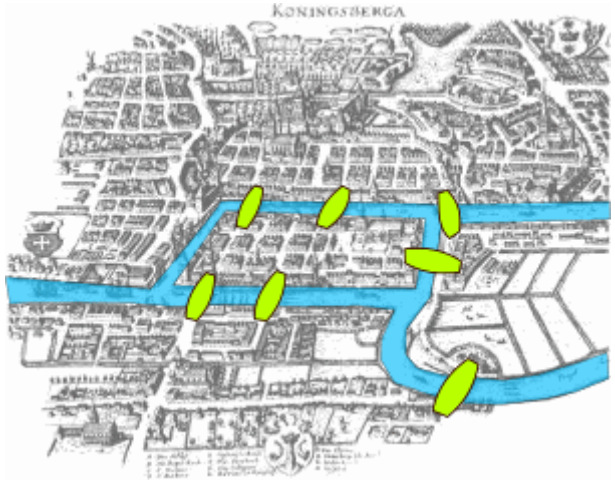
- Graph
  - An abstract representation of a set of objects where some pairs of the objects are connected by links
    - The interconnected objects are called **vertices**
    - The links that connect some pairs of vertices are called **edges**

$G = (V, E)$



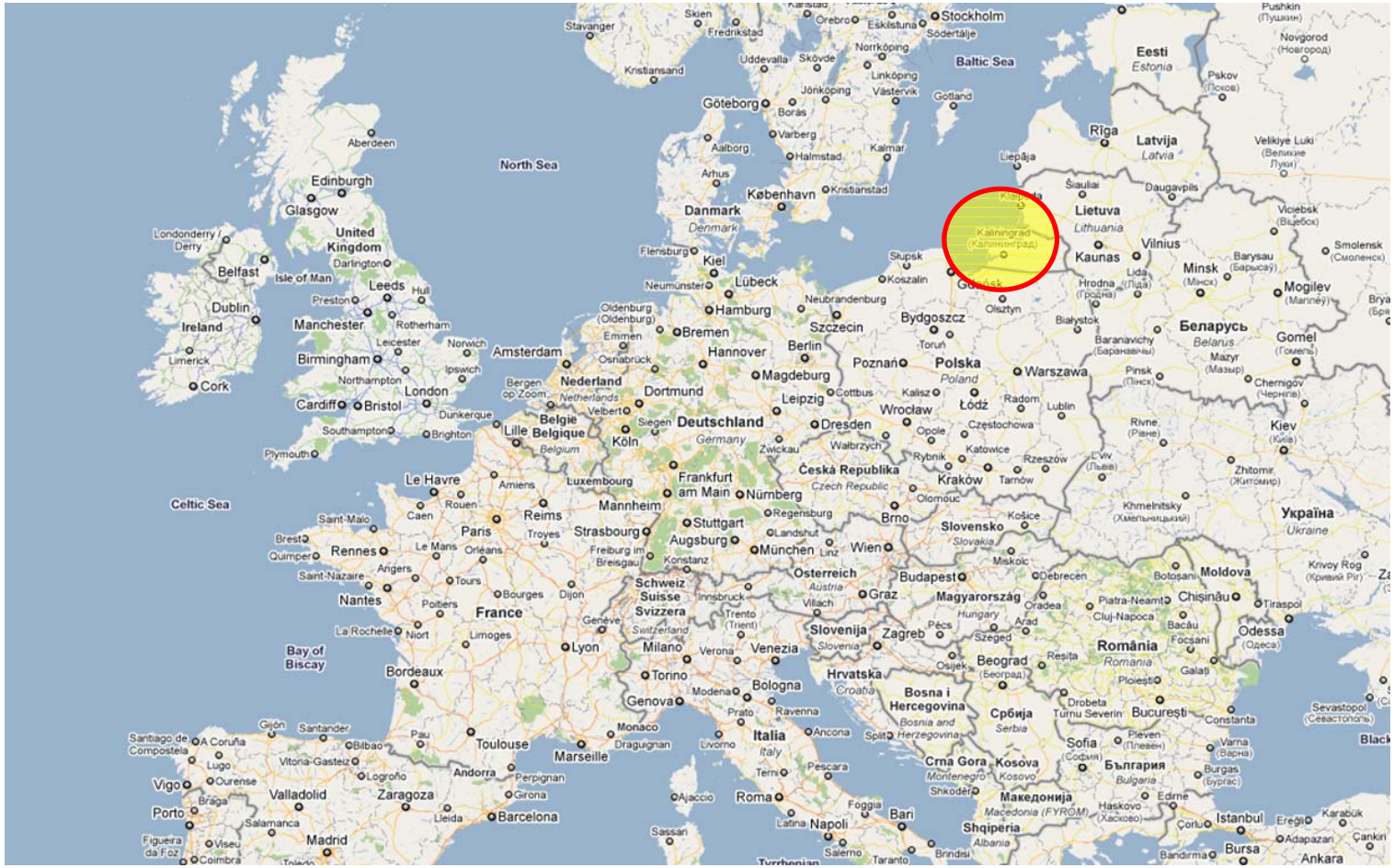
4.0 Introduction

- Konigsberg bridge problem
  - To find a walk through the city that would cross each bridge once and only once
  - Euler was invited to attack!!



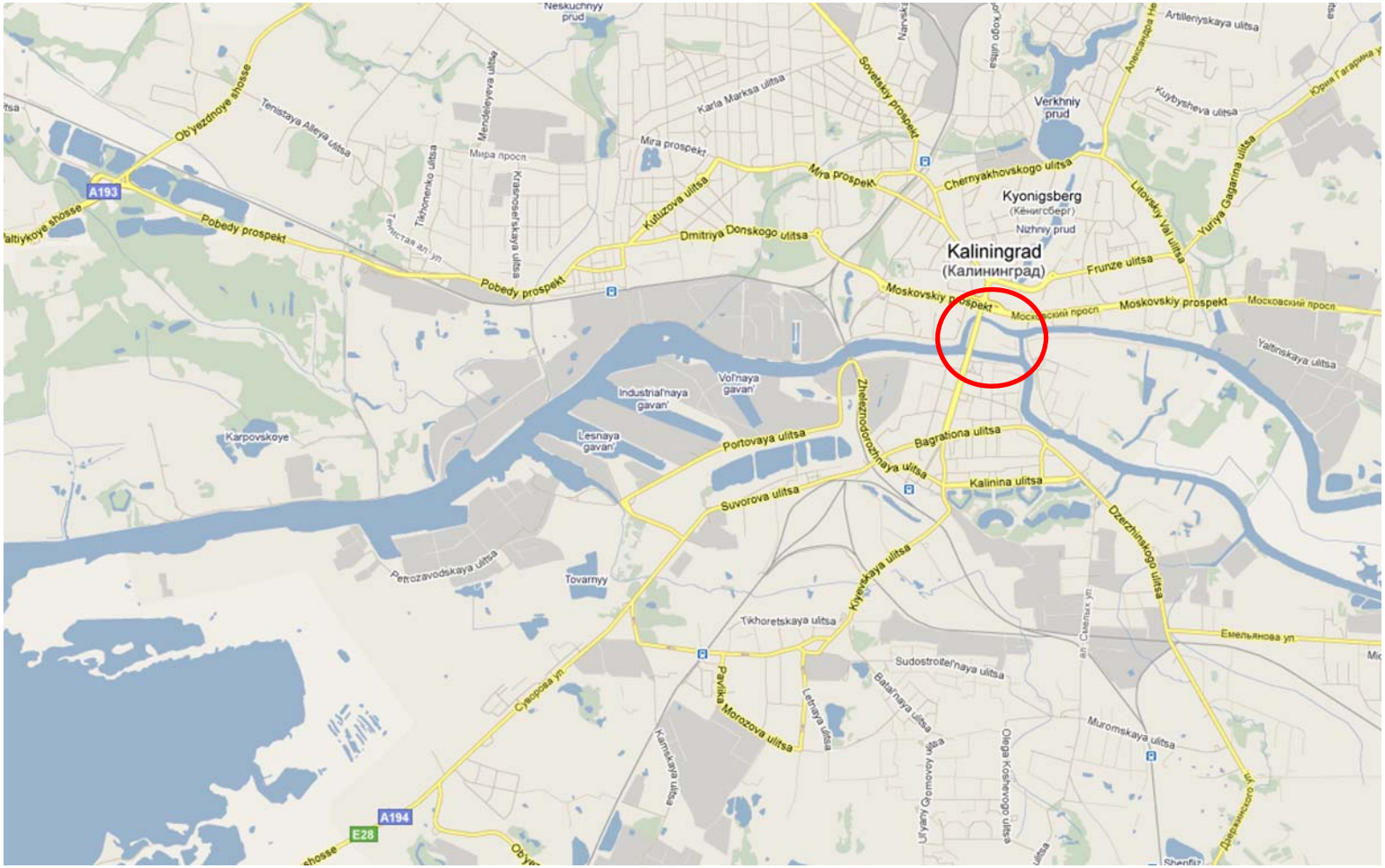
4.0 Introduction

- Konigsberg bridge problem



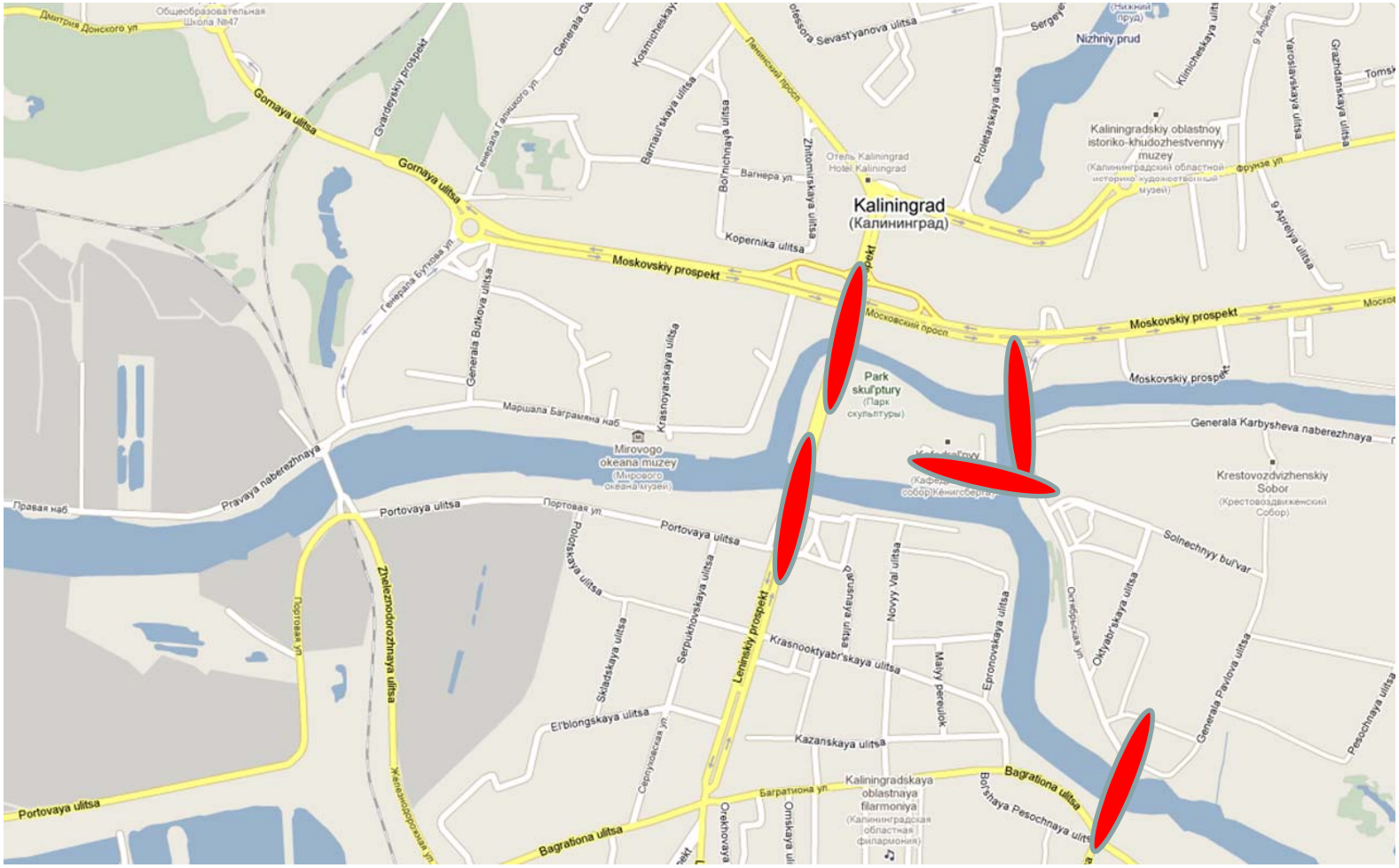
4.0 Introduction

- Konigsberg bridge problem



4.0 Introduction

- Konigsberg bridge problem



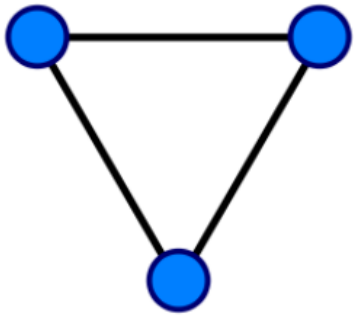
4.0 Introduction

- Euler's solution doesn't matter
- Euler introduced "graph" to explain his solution
  - Abstraction
    - Land & island → node (vertex)
    - Bridge → edge (link)

4.1 What is graph?

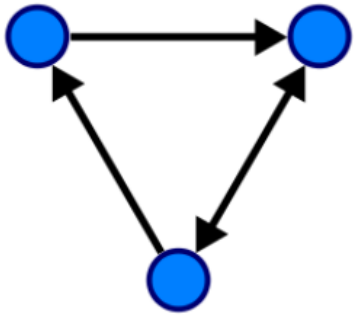
(1) Definition of a graph

- specified by a set of vertices (also called *nodes*) *V* and by edges *E* between select pairs of vertices.
- G = (V, E)
- Ex: G in the previous page
  - V = {1, 2, …, 13}
  - E = { {1, 2}, {1, 3}, … }



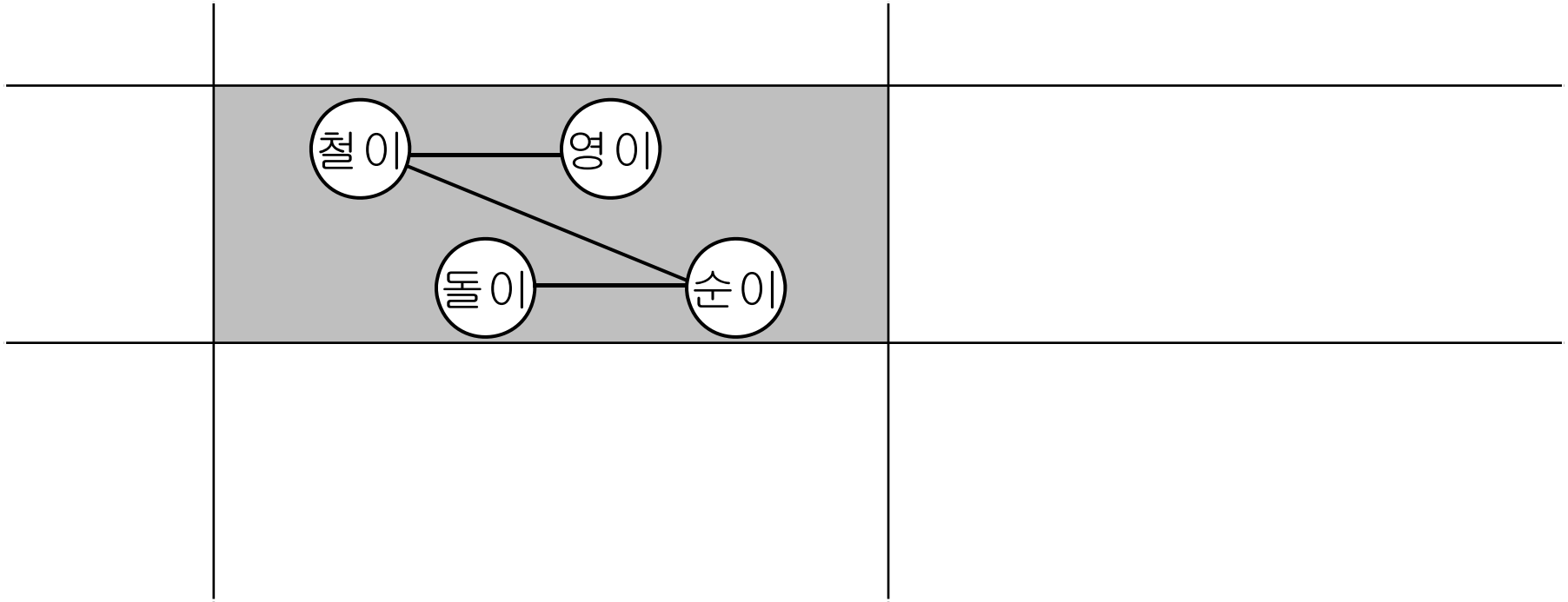
(2) The type of a graph

- An undirected graph
  - {v, w} = {w, v}
- A directed graph
  - (v, w) ≠ (w, v)



4.1 What is graph?

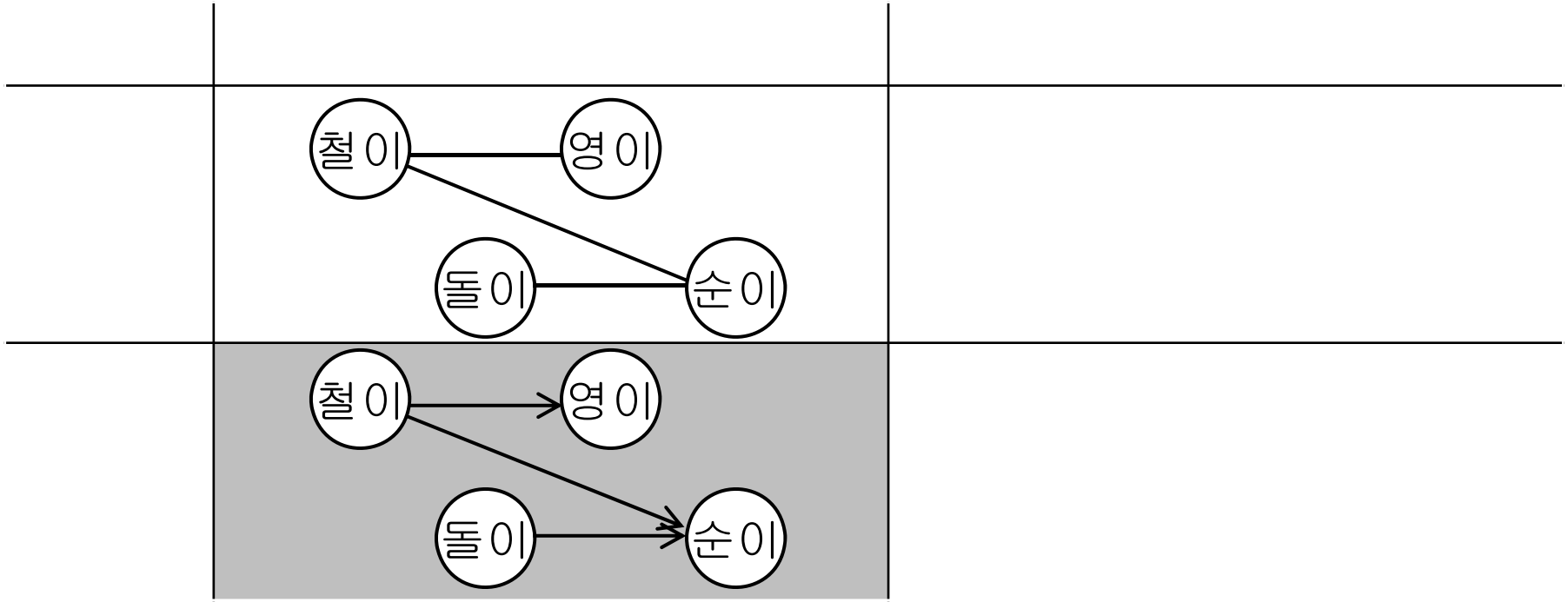
(2) The type of a graph



개체: 철이, 영이, 돌이, 순이
관계: 철이, 영이는 페친임.
철이, 순이는 페친임.
돌이, 순이는 페친임.

4.1 What is graph?

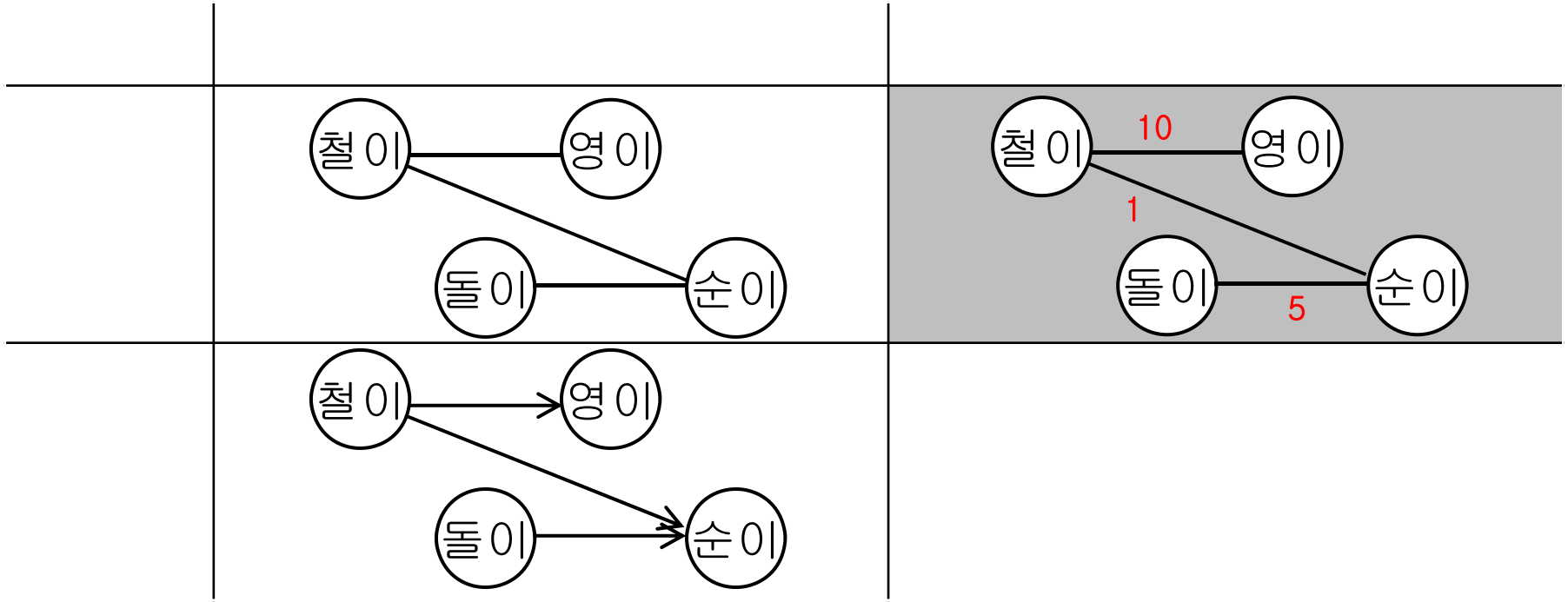
(2) The type of a graph



개체: 철이, 영이, 돌이, 순이
관계: 철이는 영이를 follow함.
철이는 순이를 follow함.
돌이는 순이를 follow함.

4.1 What is graph?

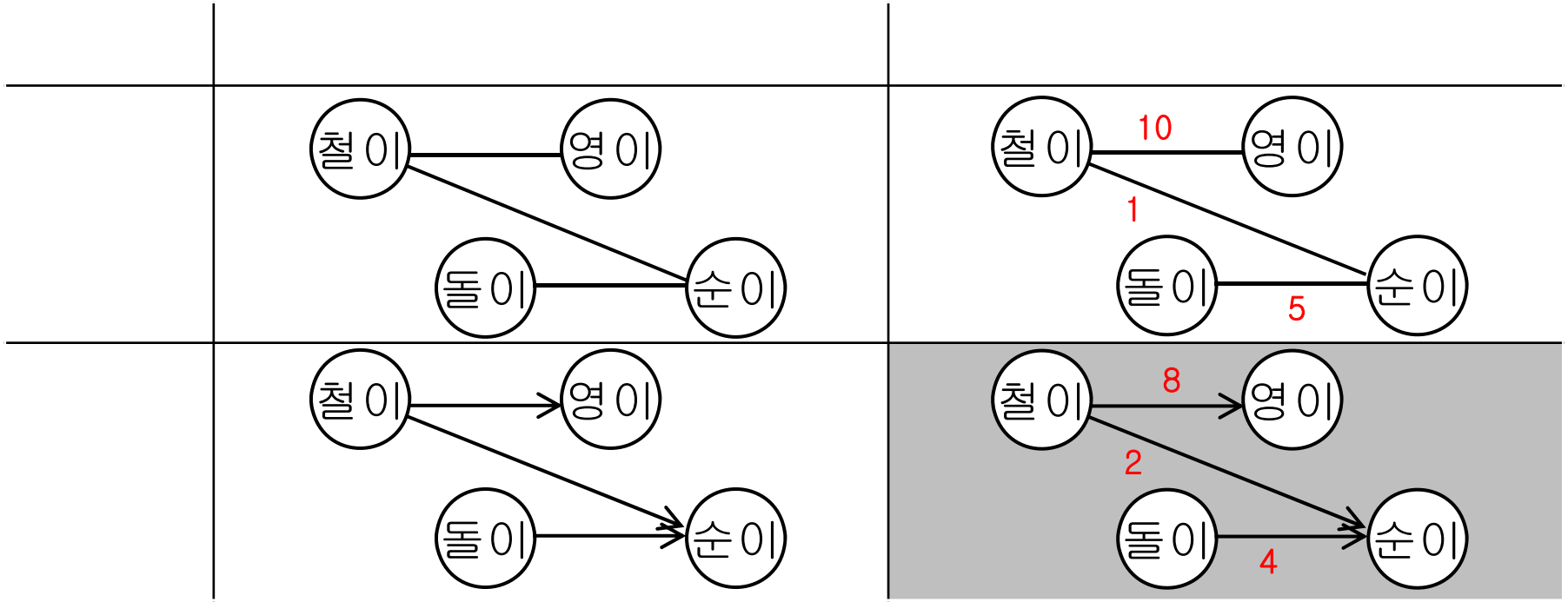
(2) The type of a graph



개체: 철이, 영이, 돌이, 순이
관계: 철이, 영이는 심각한 페친임.
철이, 순이는 썰렁한 페친임.
돌이, 순이는 평범한 페친임.

4.1 What is graph?

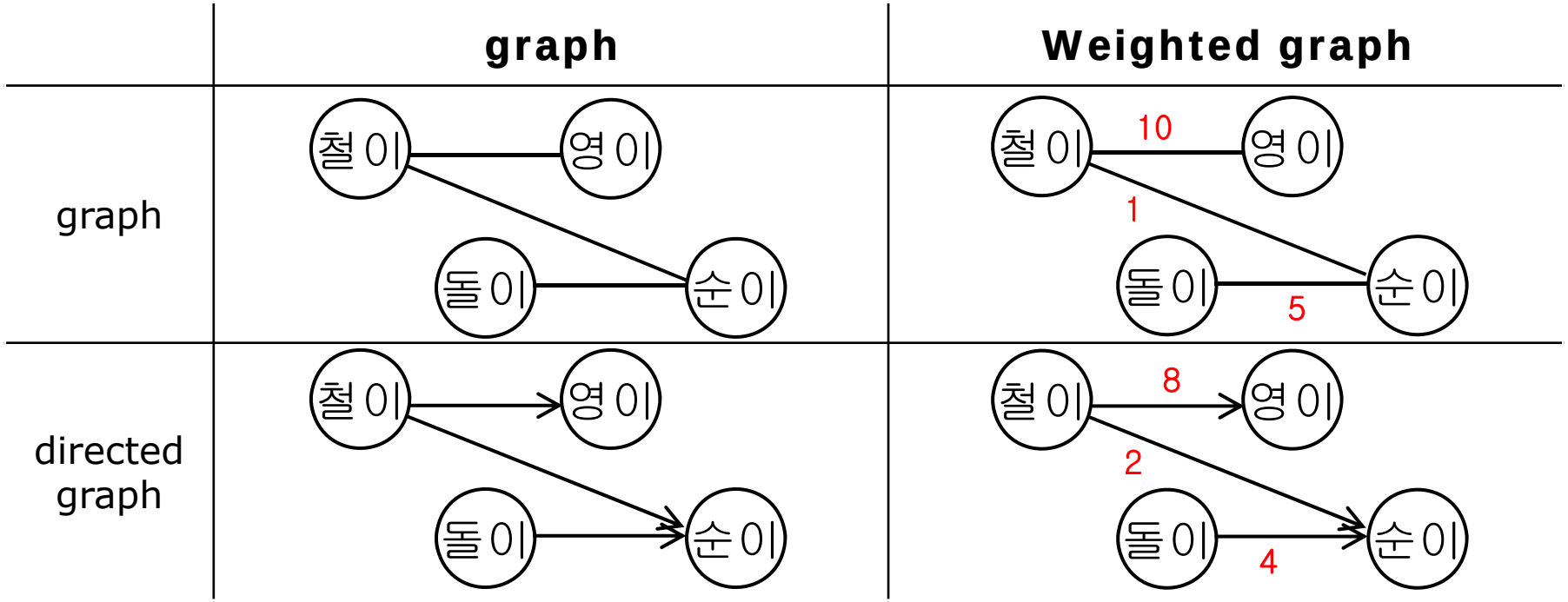
(2) The type of a graph



개체: 철이, 영이, 돌이, 순이
관계: 철이는 영이를 많이 follow함.
철이는 순이를 조금 follow함.
돌이는 순이를 보통 follow함

4.1 What is graph?

(2) The type of a graph

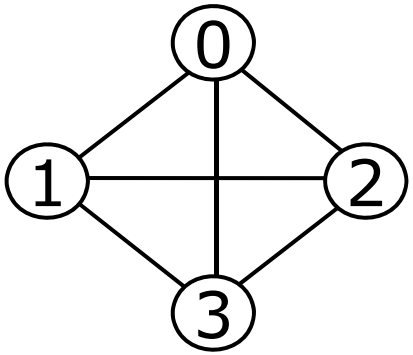
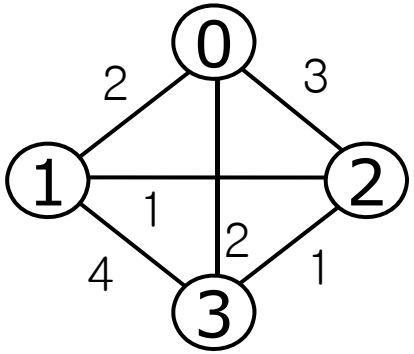
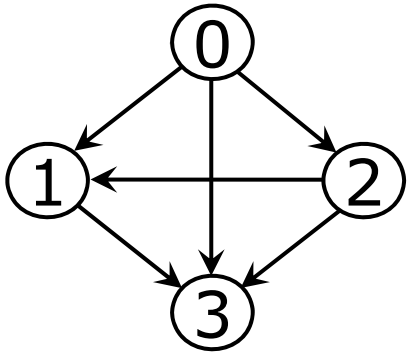
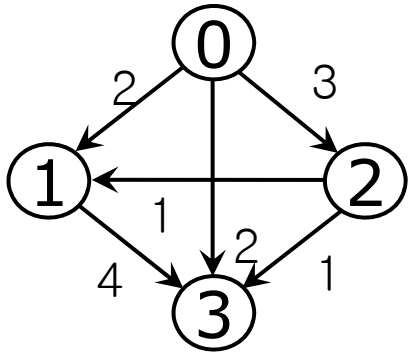


(Undirected) Graph: (u, v) = (v, u)

Directed graph: (u, v) ≠ (v, u)

4.1 What is graph?

(2) The type of a graph

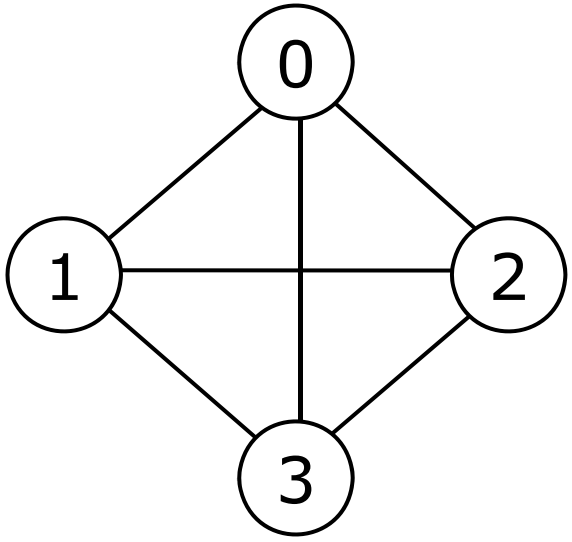
		Weighted or not	
		Non-weighted	**Weighted**
Directed or not	**Undirected**		
	Directed		

4.1 What is graph?

(3) Representation of a graph

(3.1) Edge list

- A list of edges
- Available on many coding problems



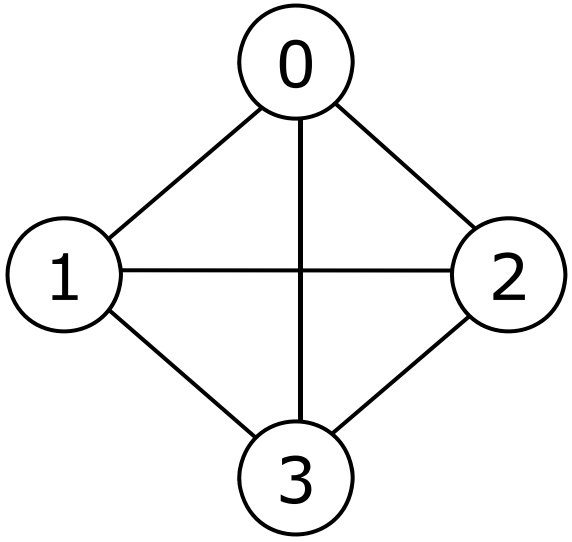
4, 6
0, 1
0, 2
0, 3
1, 2
1, 3
2, 3

4.1 What is graph?

(3) Representation of a graph

(3.2) Adjacency matrix of G = (V, E)

- A two-dimensional n X n array: a[n][n]
- a[i][j] = 1, if $(v_i, v_j) \in E$
- a[i][j] = 0, if $(v_i, v_j) \notin E$



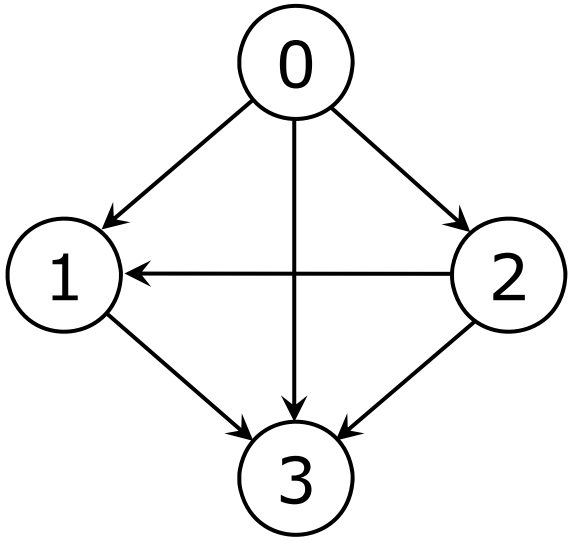
$$\begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \end{array} \begin{array}{c} \begin{array}{cccc} 0 & 1 & 2 & 3 \end{array} \\ \begin{bmatrix} & & & \\ & & & \\ & & & \\ & & & \end{bmatrix} \end{array}$$

4.1 What is graph?

(3) Representation of a graph

(3.2) Adjacency matrix of G = (V, E)

- A two-dimensional n X n array: a[n][n]
- a[i][j] = 1, if $<v_i, v_j> \in E$
- a[i][j] = 0, if $<v_i, v_j> \notin E$



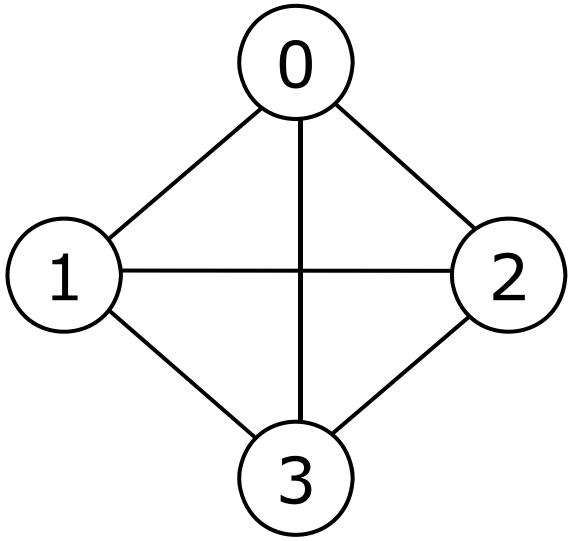
$$\begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \end{array} \begin{array}{c} \begin{array}{cccc} 0 & 1 & 2 & 3 \end{array} \\ \begin{bmatrix} & & & \\ & & & \\ & & & \\ & & & \end{bmatrix} \end{array}$$

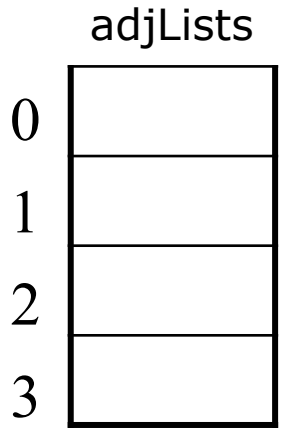
4.1 What is graph?

(3) Representation of a graph

(3.3) Adjacency list of G = (V, E)

- adjLists[n]
- adjLists[i] is a pointer to the first node in the adjacency list for vertex i



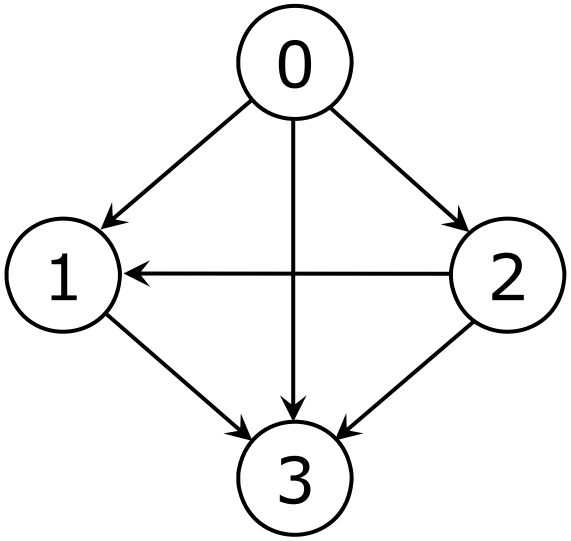


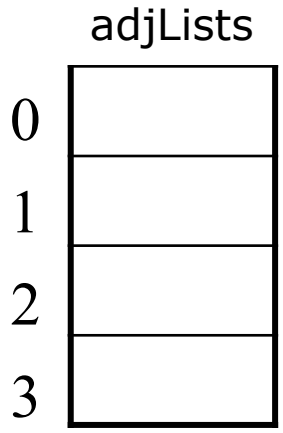
4.1 What is graph?

(3) Representation of a graph

(3.3) Adjacency list of G = (V, E)

- adjLists[n]
- adjLists[i] is a pointer to the first node in the adjacency list for vertex i

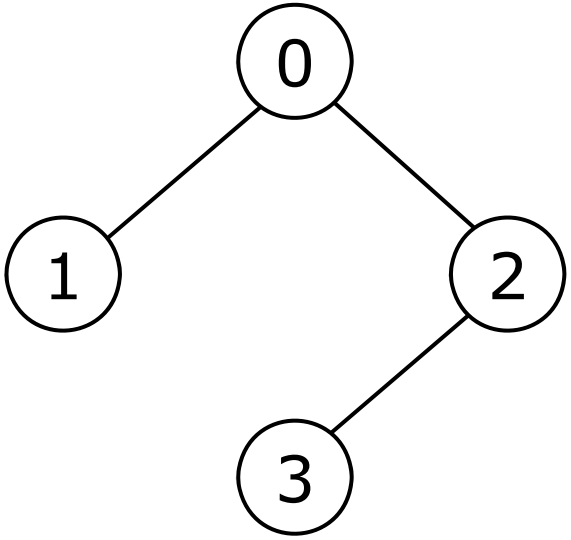




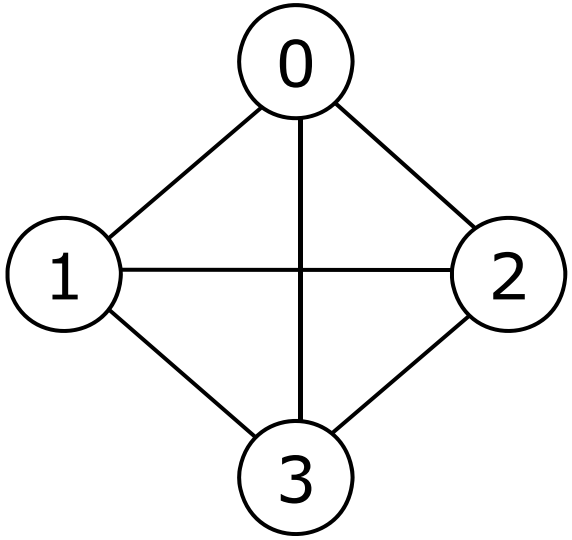
4.1 What is graph?

(4) Performance analysis

(4.1) Sparse VS dense (complete) graph



Sparse graph:
|V| = n
|E| = O(n)



Complete graph:
|V| = n
|E| = $O(n^2)$

4.1 What is graph?

(4) Performance analysis

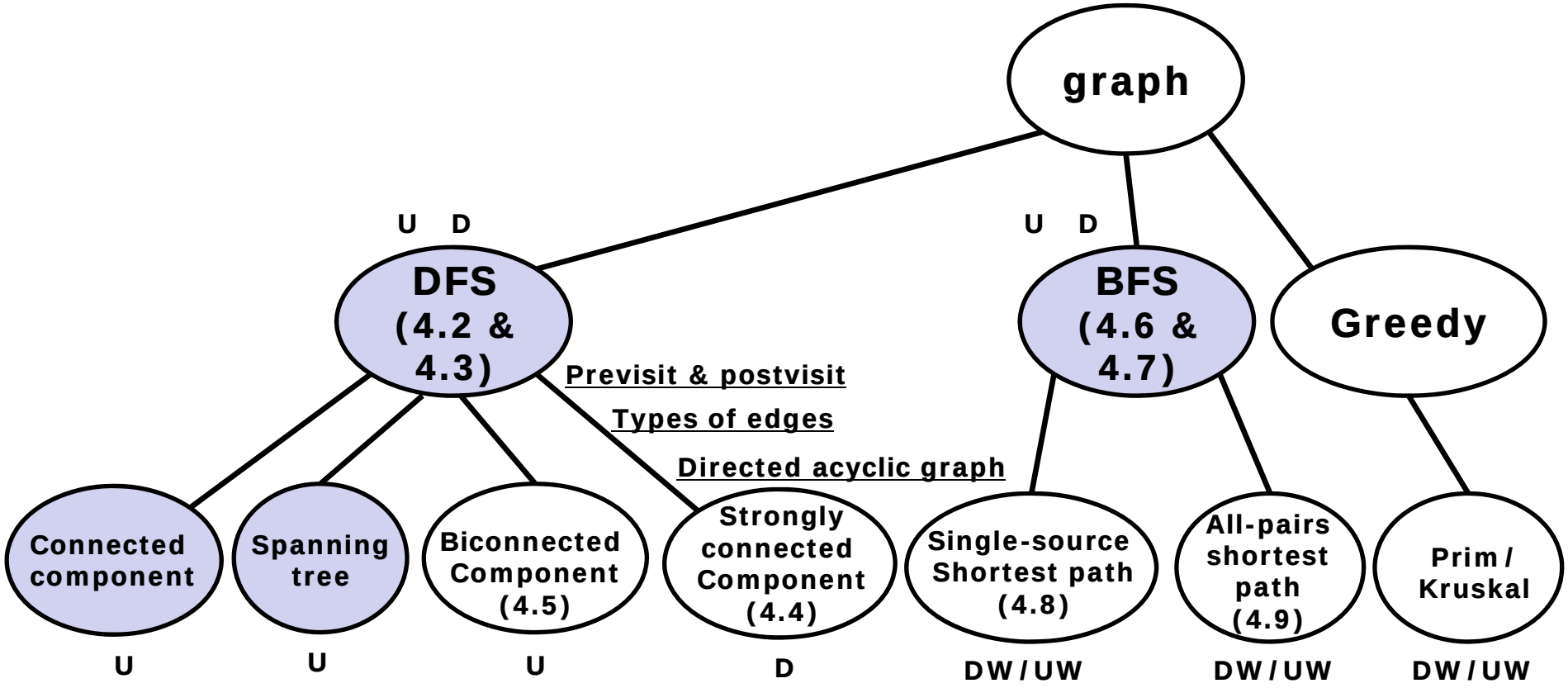
(4.2) Performance analysis

Space complexity	Sparse graph	Complete graph
Adjacency list	**O(n)**	$O(n^2)$
Adjacency matrix	$O(n^2)$	$O(n^2)$

Time complexity	Sparse graph	Complete graph
Adjacency list	**O(n)**	$O(n^2)$
Adjacency matrix	$O(n^2)$	$O(n^2)$

4.1 What is graph?

(5) Classification of graph algorithms



4.1 What is graph?

다음은 그래프에 대한 설명이다. 잘못된 것을 모두 고르시오.

(a) 그래프는 3개 이상의 개체들 사이의 관계를 동시에 나타낼 수 있다.

(b) 그래프를 adjacency matrix로 표현하면 sparse graph나 dense graph나 같은 기억 공간을 요구한다.

(c) 그래프는 개체, 관계, 속성의 3 개의 요소를 갖는다.

(d) 그래프의 vertex는 개체를 나타내고 edge는 속성을 나타낸다.