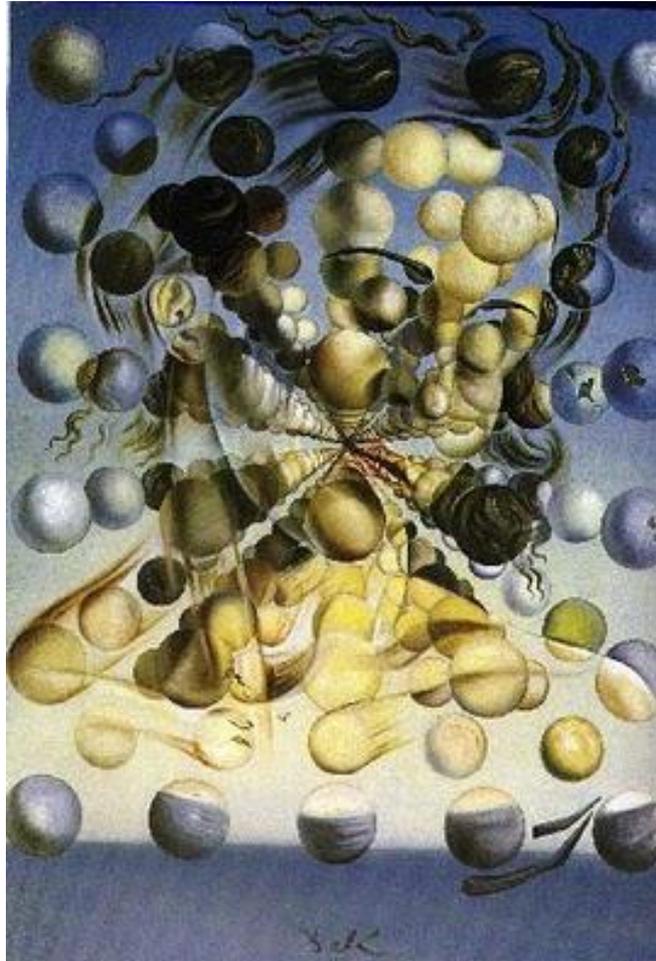# Graphs and Isomorphisms

Galatea of the
Spheres - Dali

Discrete Structures (CS 173) Lecture B (Professor Gul Agha)
Sahand Mozaffari
Based on slides by Derek Hoiem, University of Illinois
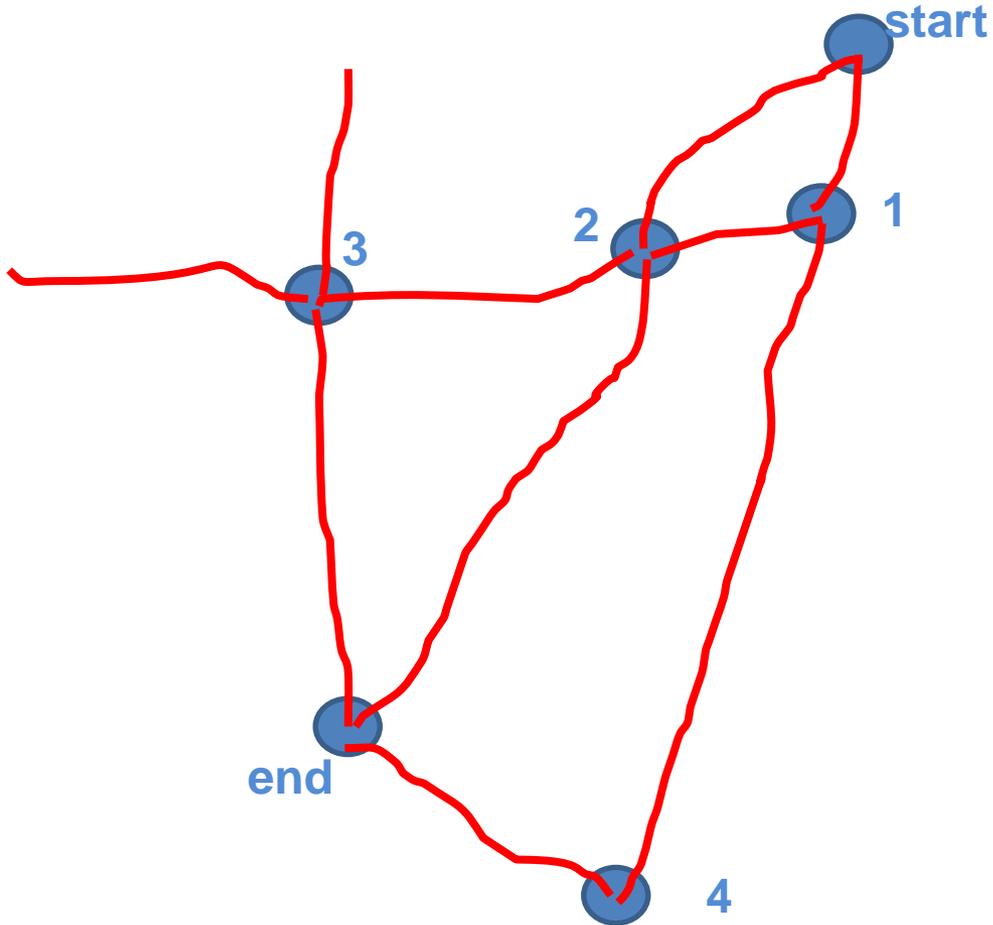
Today's Lecture: Graphs

- How to represent graphs?

- What are the properties of a graph?
  - Degrees, special types

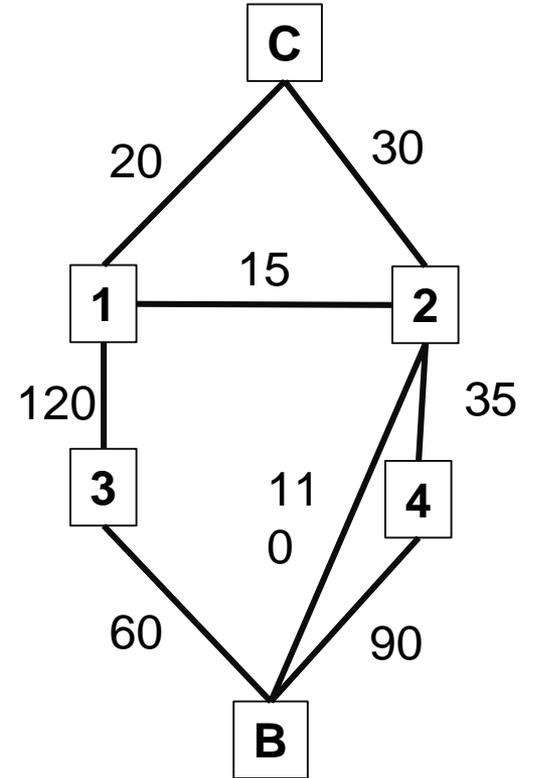- When are two graphs isomorphic, having the same structure?

Fastest path from Chicago to Bloomington?

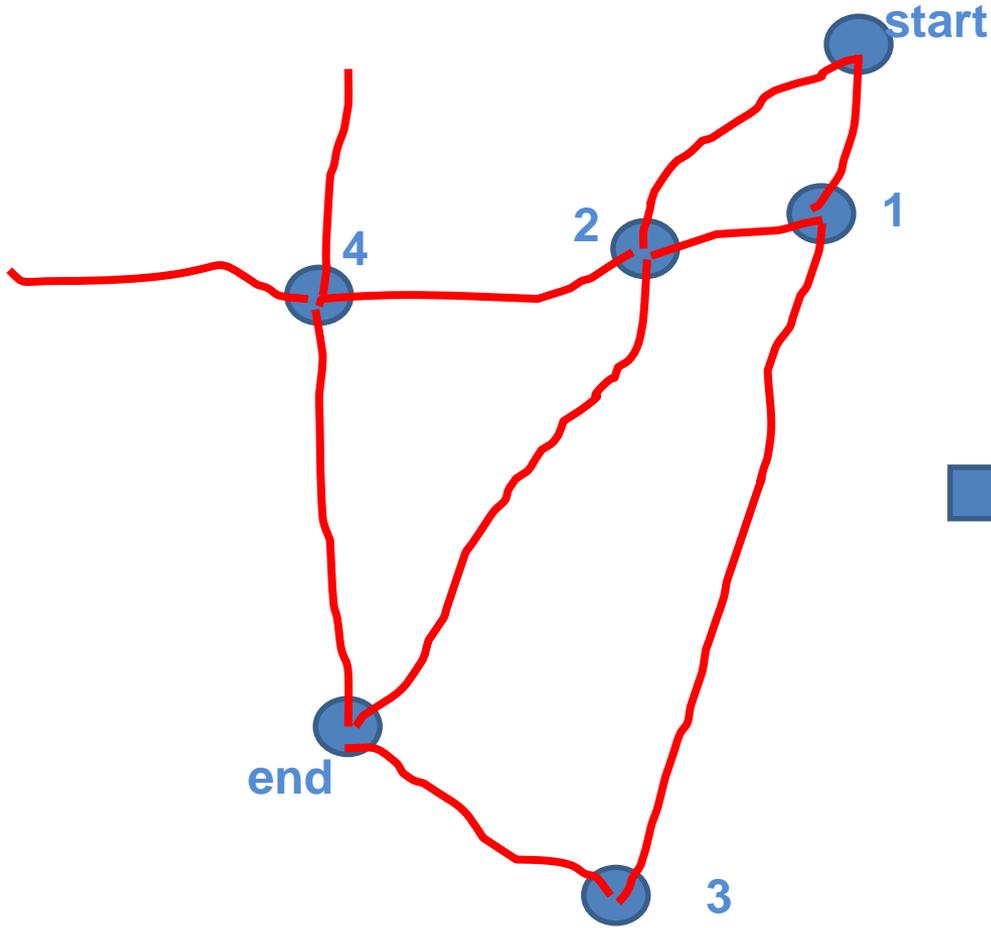# Fastest path from Chicago to Bloomington?

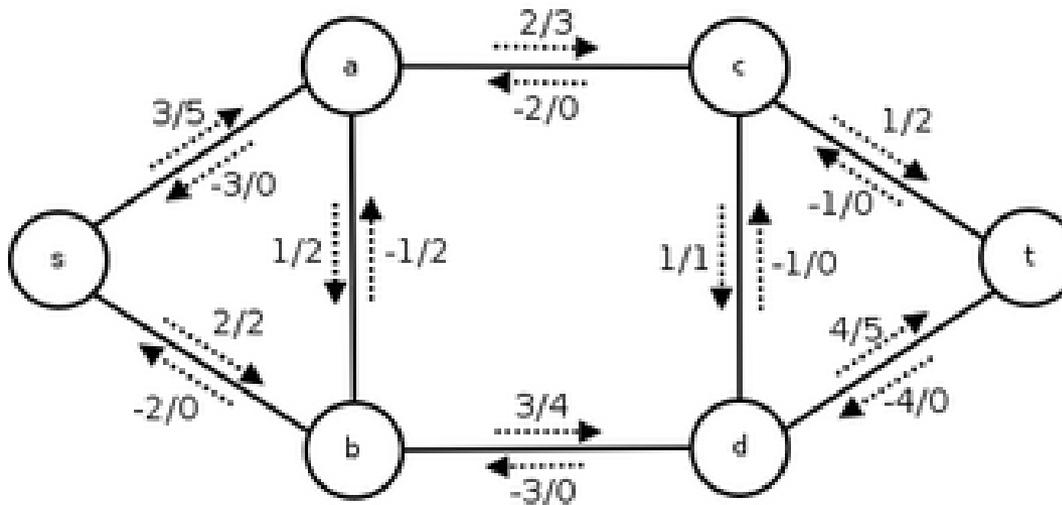# Fastest path from Chicago to Bloomington?

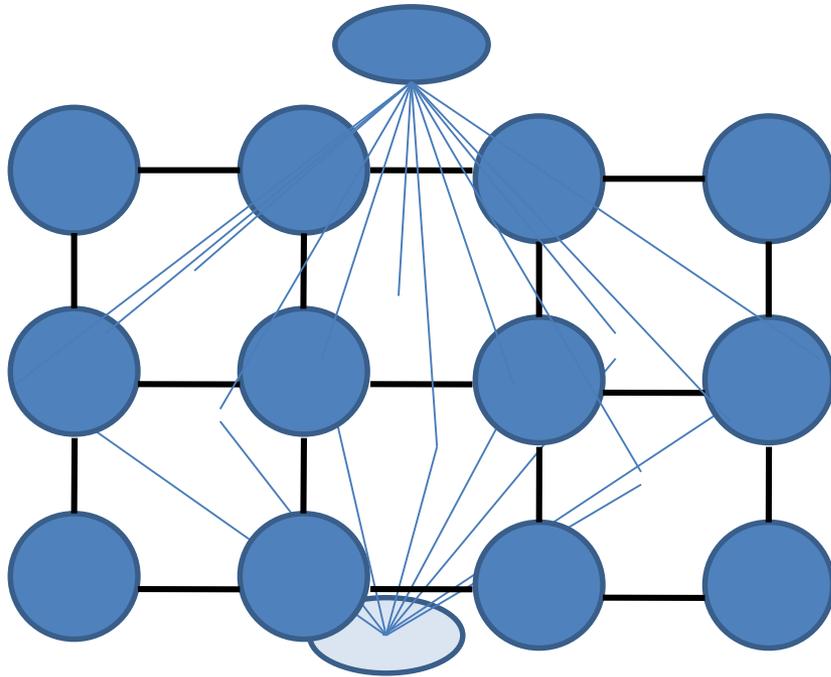# Fastest path from Chicago to Bloomington?

Other applications of graphs

- Modeling the flow of a network
  - Traffic, water in pipes, bandwidth in computer networks, etc.

Other applications of graphs

- Modeling probabilistic dependencies

Graphical model of dependencies between pixels for object segmentation

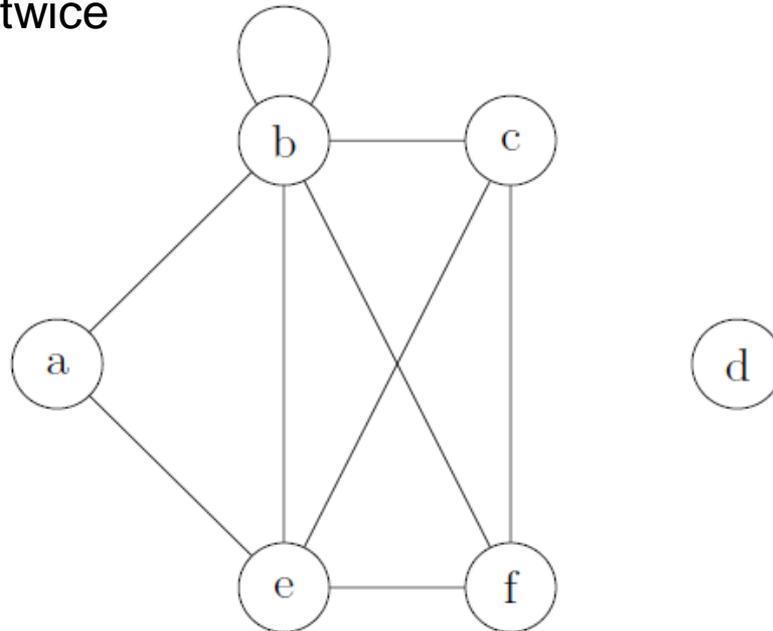Segmented object with "GrabCut" by Rother et al. 2004

Basics of graphs

Graph = (V, E)

Terminology: vertex/node, edge, neighbor/adjacent, directed vs. undirected, simple graph, degree of a node

overhead

Degrees and handshaking theorem
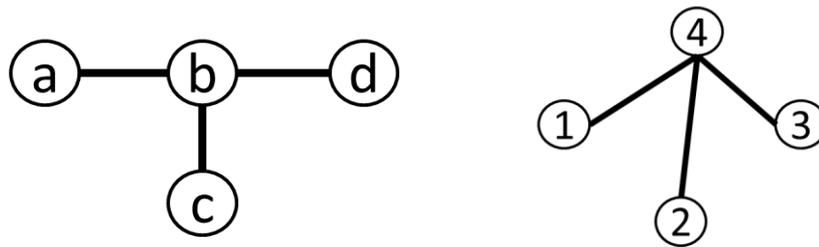
$$\sum_{v \in V} deg(v) = 2|E|$$

Loops count twice

overhead

Isomorphism

An **isomorphism** from $G_1 = (V_1, E_1)$ to $G_2 = (V_2, E_2)$ is a bijection $f: V_1 \to V_2$ such that any pair of nodes a and b are joined by an edge iff $f(a)$ and $f(b)$ are joined by an edge

Two graphs are **isomorphic** if there is an isomorphism between them

overhead

Isomorphism examples

An **isomorphism** from $G_1 = (V_1, E_1)$ to $G_2 = (V_2, E_2)$ is a bijection $f : V_1 \to V_2$ such that any pair of nodes a and b are joined by an edge iff $f(a)$ and $f(b)$ are joined by an edge
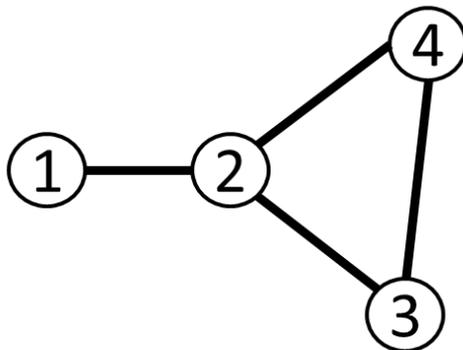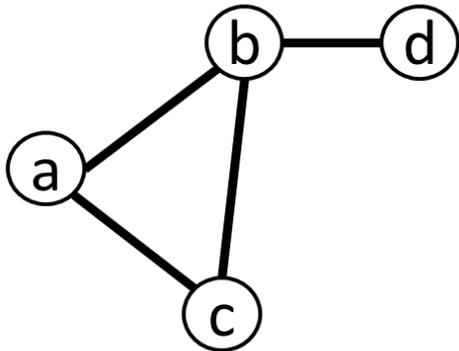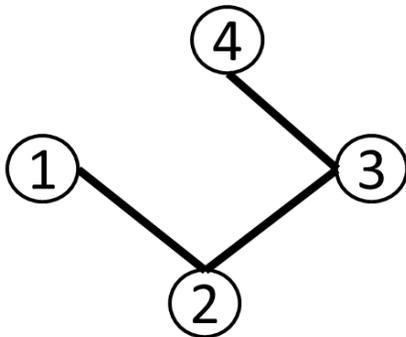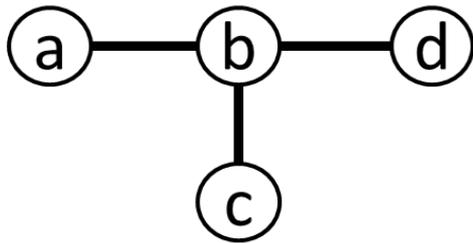
overhead

Isomorphism examples

An **isomorphism** from $G_1 = (V_1, E_1)$ to $G_2 = (V_2, E_2)$ is a bijection $f: V_1 \rightarrow V_2$ such that any pair of nodes a and b are joined by an edge iff $f(a)$ and $f(b)$ are joined by an edge

overhead

Requirements for graphs to be isomorphic

overhead

Requirements for two graphs to be isomorphic

- Same number of nodes and edges

- Same number of nodes of degree k

- Every subgraph in the first must have a matching subgraph in the second

● Degrees of the neighbors should be the same
● Distance between vertices should be the same

Types of graphs

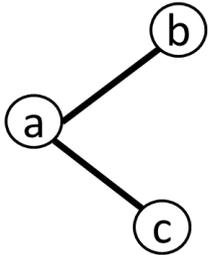K_n: complete graph with n nodes

C_n: cycle graph with n nodes

W_n: wheel graph with $n + 1$ nodes

How many edges does each type have?

overhead

Practice identifying types of graphs

Automorphism: an isomorphism from a graph to itself

- Automorphisms identify symmetries in the graph

- How many different automorphisms?

C_4

C_6

Smallest graph without non-trivial automorphism?

Isomorphism is an equivalence relation: reflexive, symmetric, and transitive

Isomorphisms

How many graphs are there on a given set of n vertices?

How many isomorphic equivalence sets in three node graphs?

overhead

Things to remember

- A graph is defined by a set of nodes and a set of edges that connect them

- Be able to identify types of graphs and degrees of nodes

- Be able to identify isomorphisms (or lack thereof) between graphs