Graphs
City of Königsberg, Prussia, 1735.

**Task:** Find a path through the city that would cross each bridge once and only once.
Task: Find a path through the city that would cross each bridge once and only once.
Leonhard Euler
1707–1783

e - k + f = 2

130

Angelo Boog
2007
Task: Find a path through the city that would cross each bridge once and only once.
**Basic definitions**

**Def.** *Graph G = (V, E)* is a set of *vertices V*, with a set of *edges E* between them.

**Def.** Each edge has *two endpoints*.

**Def.** An edge *joins* its endpoints, two endpoints are *adjacent* if they are joined by an edge.

**Def.** An edge is said to be *incident* to the vertices it joins.
Basic definitions

V = \{a, b, c, d, e, f\}

E = \{\{a, b\}, \{a, c\}, \{a, d\}, \{c, d\}, \{c, f\}\}
Subgraphs

Deleting some vertices or edges from a graph leaves a subgraph. Formally:

**Def.** A subgraph of $G = (V, E)$ is a graph $G' = (V', E')$ where $V'$ is a nonempty subset of $V$ and $E'$ is a subset of $E$.

- $V' = \{a, c, f, e\}$
- $E' = \{\{a, c\}, \{c, f\}\}$
Variants: Multigraph

**Def.** In *simple graphs*, each pair of distinct vertices has at most one edge.

**Def.** Graphs that may have multiple edges connecting the same vertices are called *multigraphs*.
Variants: Graphs with loops

Some graphs that may include loops, and possibly multiple edges connecting the same pair of vertices or a vertex to itself.
Directed graphs

**Def.** In *directed graph* (or digraph) the edges are directed, that is every edge \((u, v)\) is an ordered pair. It starts at \(u\) and ends at \(v\).
**Complete graph,** $K_n$

**Def.** *Complete graph* is a simple graph that has one edge between each pair of vertices.

They are denoted by $K_n$, where $n$ is the number of vertices.

$K_6$ is in the figure above.
Empty graph

Def. *Empty graph* has empty set of edges.
Degree in undirected graphs

**Def.** The *degree* of a vertex in an undirected graph is the number of edges incident with it, except that a loop at a vertex contributes twice to the degree of that vertex.

The degree of the vertex $v$ is denoted by $\text{deg}(v)$.

$\text{deg}(a) = 7, \quad \text{deg}(b) = 2, \quad \text{deg}(c) = 4,$
$\text{deg}(d) = 4, \quad \text{deg}(e) = 0, \quad \text{deg}(f) = 1.$
The handshaking lemma

**Lemma** (The handshaking lemma). Let \((V, E)\) be an undirected graph with \(m\) edges. Then

\[
\sum_{v \in V} \deg(v) = 2m.
\]

**Corollary.** An undirected graph has an even number of vertices of odd degree.
Social graphs

1. Prove that there is no group of 7 people such that each person in the group has exactly 3 friends in the group.

 Friendship is always mutual.
That is, in math-speak, the *friendship relationship is symmetric*.

2. Then, try to prove that in any group of $n \geq 2$ people, there are at least 2 people with the same number of friends in the group.
Degree in directed graphs

**Def.** In directed graphs, there are similar notions of *in-degree* and *out-degree*, denoted by $\deg^-(v)$ and $\deg^+(v)$ respectively

\[
\begin{align*}
\deg^-(a) &= 3, & \deg^+(a) &= 3, & \deg^-(b) &= 1, & \deg^+(b) &= 1, \\
\deg^-(c) &= 1, & \deg^+(c) &= 3, & \deg^-(d) &= 2, & \deg^+(d) &= 1, \\
\deg^-(e) &= 0, & \deg^+(e) &= 0, & \deg^-(f) &= 1, & \deg^+(f) &= 0.
\end{align*}
\]
Theorem. Let \((V, E)\) be a directed graph. Then

\[
\sum_{v \in V} \deg^{-}(v) = \sum_{v \in V} \deg^{+}(v) = |E|.
\]
Matching

**Def.** A *matching* $M$ in a simple graph $(V, E)$ is a subset of $E$ such that no two edges from $M$ are incident with the same vertex.

In other words, a matching is a set of *disjoint* edges.

Also, we can introduce maximum, maximal, perfect matchings.
Def. A simple graph is called bipartite if its vertex set $V$ can be partitioned into two disjoint sets $V_1$ and $V_2$ such that every edge in the graph connects a vertex in $V_1$ and a vertex in $V_2$. 
Matching

Suppose that there are $m$ employees in a group and $n$ different jobs that need to be done, where $m \geq n$. 

![Graph representation of matching problem]

- Alice
- Bob
- Charlie
- David
- Eve
- Frank
- 3D modeling
- Textures
- Programming graphics
- Programming
- Music
Matching

Suppose that there are $m$ employees in a group and $n$ different jobs that need to be done, where $m \geq n$. 

[Diagram of matching between employees and jobs]
**Complete matching from $V_1$ to $V_2$**

**Def.** We say that a matching $M$ in a bipartite graph $G = (V, E)$ with bipartition $(V_1, V_2)$ is a *complete matching from $V_1$ to $V_2$* if every vertex in $V_1$ is the endpoint of an edge in the matching, or equivalently, if $|M| = |V_1|$.

So, every job is assigned to some employee, and no employee is assigned to more than one job.
Neighborhood of a vertex

\[ N(\{b\}) = \{f, d, c\} \]
Neighborhood of a set of vertices

Given a set of vertices $S$, define $N(S)$ to be the set of all neighbors of $S$; that is, all vertices that are adjacent to a vertex in $S$, but not actually in $S$.

$$N(\{1, 2\}) = \{a, b, c\}$$
Neighborhood of a set of vertices

Given a set of vertices $S$, define $N(S)$ to be the set of all neighbors of $S$; that is, all vertices that are adjacent to a vertex in $S$, but not actually in $S$.

\[
N(\{2, 3, 4\}) = \{c, d\}
\]
Hall’s theorem

**Theorem** (Hall’s Marriage Theorem). The bipartite graph \((V, E)\) with bipartition \((V_1, V_2)\) has a complete matching from \(V_1\) to \(V_2\) if and only if

\[
|N(A)| \geq |A|
\]

for all subsets \(A \subseteq V_1\).

**Question:** Is there a complete matching from \(V_1 = \{1, 2, 3, 4\}\) to \(V_2 = \{a, b, c, d\}\)?
Graph coloring and bipartite graphs

Graph coloring is a task to assign colors to each vertex of the graph so that no two adjacent vertices are assigned the same color.

**Def.** A graph $G$ is *$k$-colorable* if each vertex can be assigned one of $k$ colors so that adjacent vertices get different colors.

**Theorem.** A simple graph is *bipartite* if and only if it is *2-colorable*. 
Graph coloring and bipartite graphs

Graph coloring is a task to assign colors to each vertex of the graph so that no two adjacent vertices are assigned the same color.

**Def.** A graph $G$ is *$k$-colorable* if each vertex can be assigned one of $k$ colors so that adjacent vertices get different colors.

**Theorem.** A simple graph is *bipartite* if and only if it is *$2$-colorable*. 

\[ 1 \quad 2 \quad 2 \quad 1 \]
\[ 1 \quad 3 \]
Graph coloring

**Def.** The *chromatic number* of a graph is the least number of colors needed for a coloring of this graph. It’s denoted by \( \chi(G) \).

The following theorem helps to estimate the chromatic number.

**Theorem.** A graph \( G \) with maximum degree at most \( k \) is \((k + 1)\)-colorable:

\[
\max_{v \in V} (\deg(v)) \leq k \quad \rightarrow \quad G \text{ is } (k + 1)\text{-colorable}.
\]
Graph coloring

\[ \max_{v \in V} (\deg(v)) \leq k \quad \rightarrow \quad G \text{ is } (k + 1)-\text{colorable}. \]

**Proof.** The theorem can be proved by induction.

*The base case.* A graph with \(|V| = 1\) does not have edges, so the maximum degree is 0, and the graph is 1-colorable.

*Inductive step.* Assume that a graph with \(n - 1\) vertices and maximum degree at most \(k\) is \((k + 1)\)-colorable.

Now, prove that a graph with \(n\) vertices and maximum degree at most \(k\) is \((k + 1)\)-colorable …
Representing graphs

How to represent a graph in a computer program?

$n$ vertices and $m$ edges.
Representing graphs

Adjacency Matrix

2-D array $n \times n$.

$a[i, j] = 1$ if there is an edge between $i$ and $j$.

\[
\begin{array}{c|cccccc}
& 0 & 1 & 2 & 3 & 4 & 5 \\
\hline
0 & 1 & 1 & 1 & 1 & & \\
1 & 1 & 1 & 1 & & & \\
2 & 1 & 1 & & 1 & & \\
3 & & 1 & & 1 & & \\
4 & 1 & & 1 & 1 & & \\
5 & & 1 & & & & \\
\end{array}
\]

Takes $O(n^2)$ space.
Representing graphs

$n$ vertices and $m$ edges.

Adjacency List

\begin{align*}
\text{adj}(0) &= \{1, 2, 4\} \\
\text{adj}(1) &= \{0, 2, 3\} \\
\text{adj}(2) &= \{0, 1, 4\} \\
\text{adj}(3) &= \{1, 4\} \\
\text{adj}(4) &= \{0, 2, 3\} \\
\text{adj}(5) &= []
\end{align*}

Takes $O(nm)$ space.
Def. A *path* from $s$ to $t$ is a sequence of edges

$$\{x_0, x_1\}, \{x_1, x_2\}, \ldots \{x_{n-1}, x_n\},$$

where $x_0 = s$, and $x_n = t$.

Def. The *length* of a path is the number of edges in it.

$$\{e, a\} \{a, b\} \{b, d\} \{d, a\} \{a, b\} \{b, c\} \{c, f\}$$
Simple path. Cycle

**Def.** A *simple path* is a path that does not contain the same edge more than once.

**Def.** A path is called a *cycle* (or *circuit*) if its first and last vertices are the same, and its length is greater than 0.

**Def.** A *simple cycle* is a cycle that does not contain the same edge more than once.