Marriage matchmaking, map colouring . . . and other graph problems

Przemysław Gordinowicz

Institute of Mathematics,
Łódź University of Technology

Graphs & Social Networks
Lecture 1

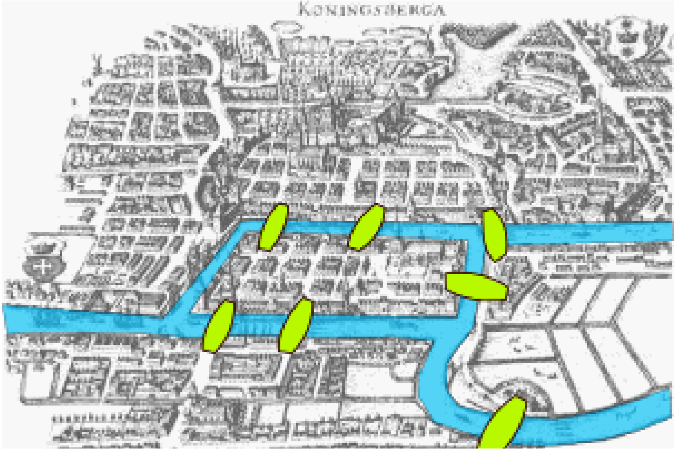
Outline

1. Graph problems
   - Bridges of Königsberg
   - Map colouring
   - Marriage matchmaking
2. What are graphs?
   - Definitions and examples
   - Selected concepts
3. Back to problems
   - Eulerian tours and trails
   - Planar graph colouring
   - Maximum matchings in bipartite graphs
4. More problems and applications

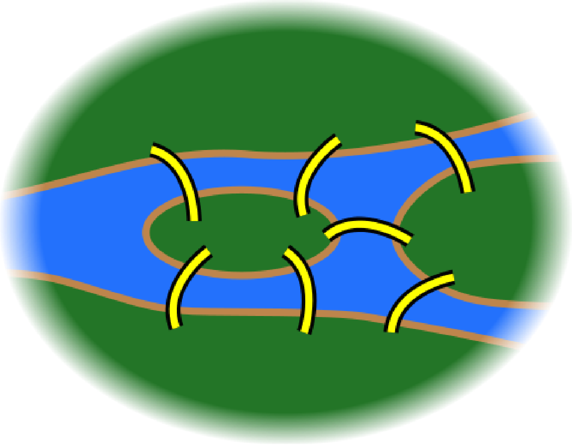
Outline

Königsberg (now Kaliningrad)

The Königsberg Bridges Problem

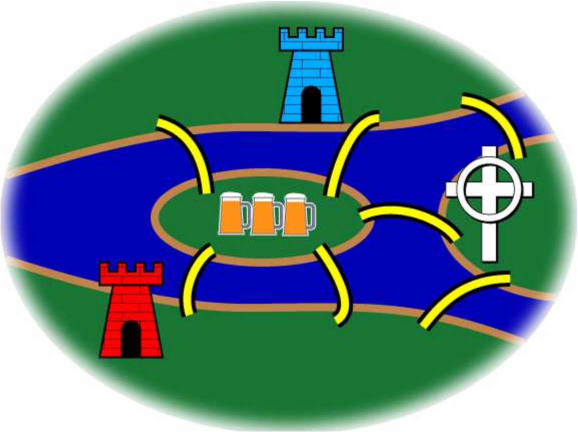


The Königsberg Bridges Problem



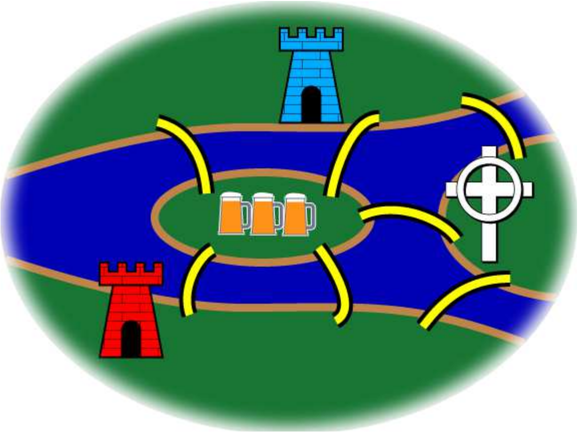
Is it possible to make such a walk, that traverses each bridge exactly once?

The Königsberg Bridges Problem



Is it possible to make such a walk, that traverses each bridge exactly once?

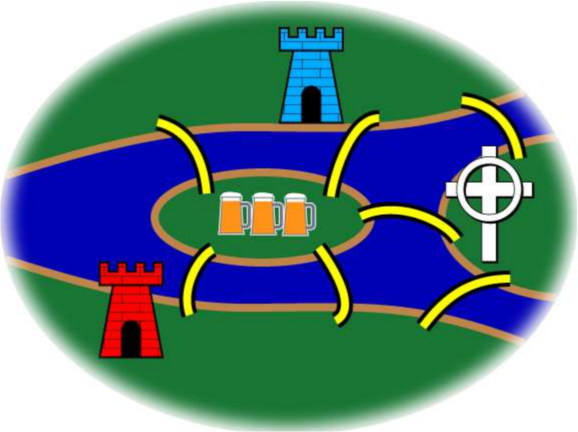
The Königsberg Bridges Problem



Is it possible to make such a walk, that traverses each bridge exactly once?

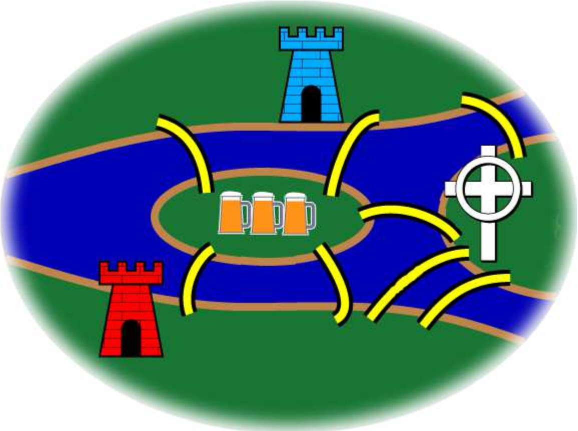
The Königsberg Bridges Problem — an answer

Leonard Euler
1735



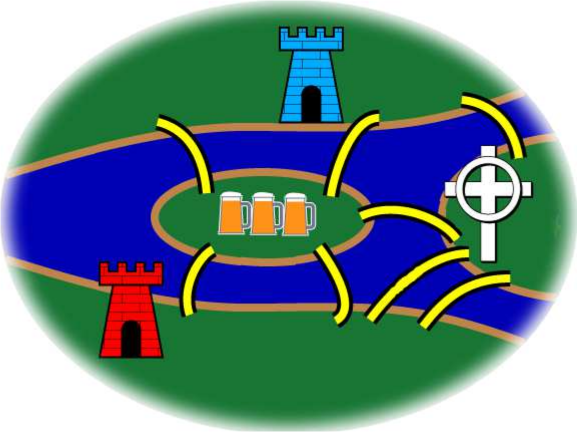
Is it possible to make such a walk, that traverses each bridge exactly once? No, it is not!

The Königsberg Bridges Problem



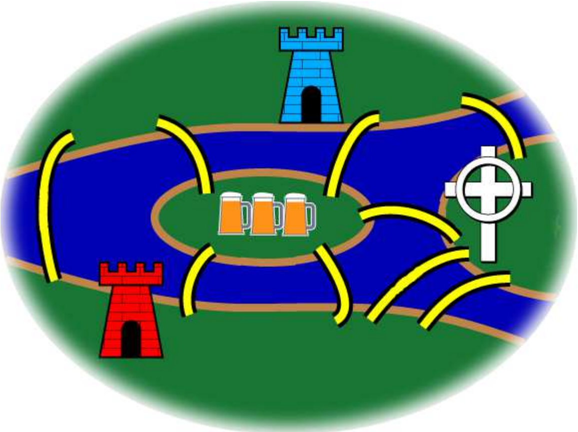
Now is it possible?

The Königsberg Bridges Problem



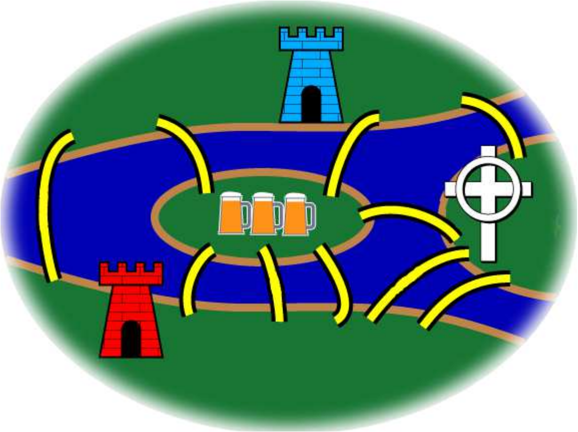
From where and to where?

The Königsberg Bridges Problem



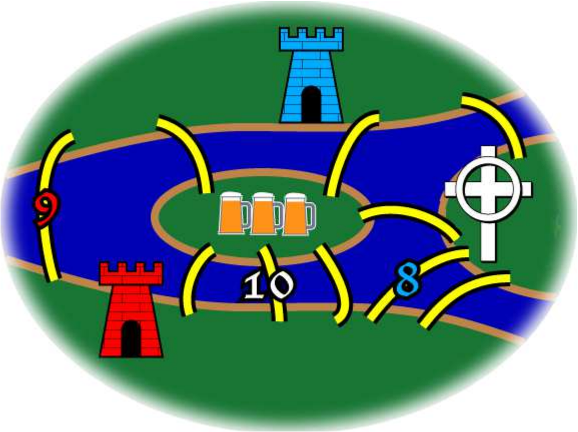
Where can you go from and to by walking once across each bridge?

The Königsberg Bridges Problem



And here?

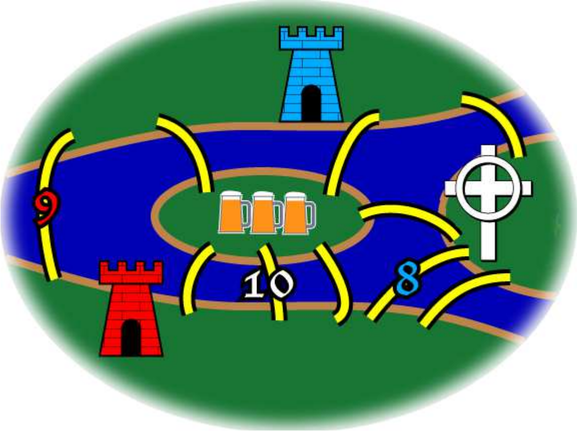
The Königsberg Bridges Problem



Eulerian circuit, Eulerian trail

The Königsberg Bridges Problem

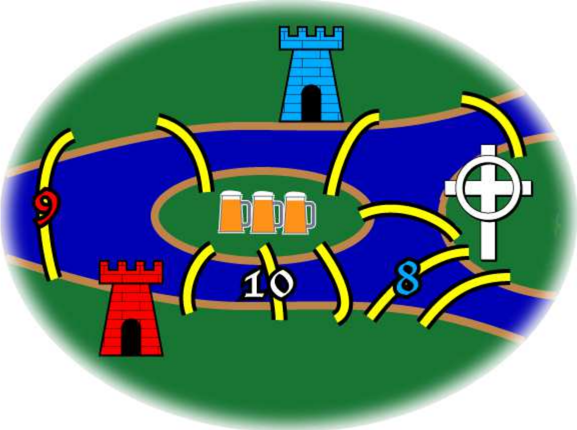
Leonard Euler



Eulerian circuit, Eulerian trail

The Königsberg Bridges Problem

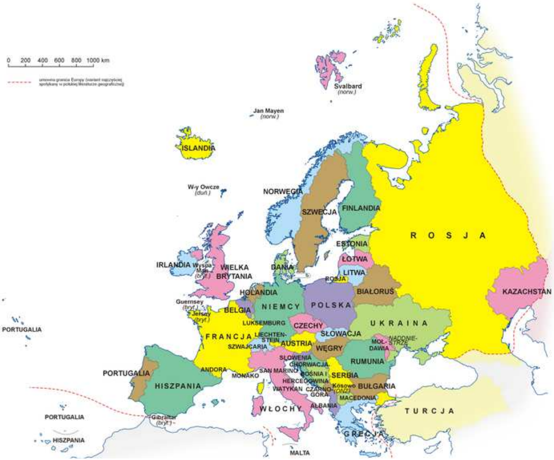
Leonard Euler



Eulerian circuit, Eulerian trail

Outline

Map colouring



Problem

How many colours do we need to colour each map?*

One important exception — Königsberg

also Alaska and Nakhchivan

Map colouring

How many colours are sufficient to colour each (connected) map?

Map colouring

Conjecture (1852)

4 colours is enough.

How many colours are sufficient to colour each (connected) map?

Map colouring

Conjecture (1852)

4 colours is enough.

Theorem (1977,1996)

Each connected map is 4-colourable.

How many colours are sufficient to colour each (connected) map?

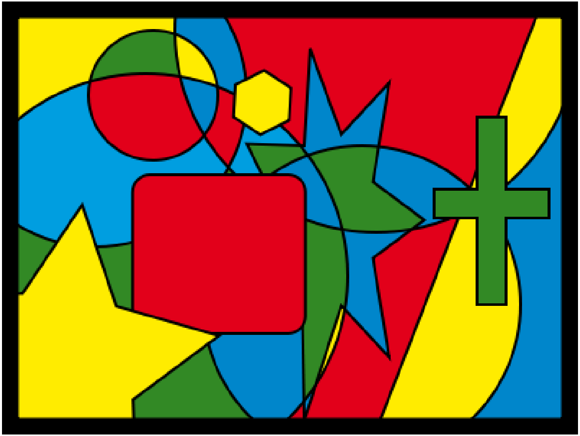
Map colouring

Conjecture (1852)

4 colours is enough.

Theorem (1977,1996)

Each connected map is 4-colourable.



How many colours are sufficient to colour each (connected) map?

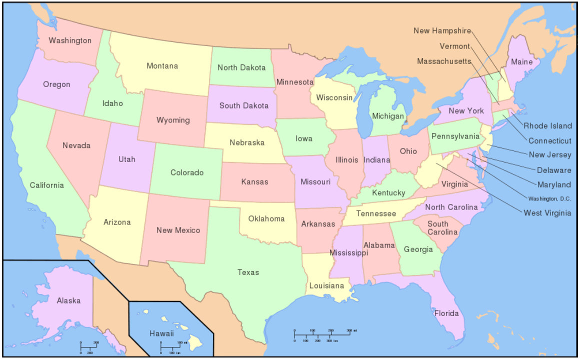
Map colouring

Conjecture (1852)

4 colours is enough.

Theorem (1977,1996)

Each connected map is 4-colourable.



How many colours are sufficient to colour each (connected) map?

Outline

Matrimonial agency problem

Question

How to maximize profits, by matching as many couples as possible?

Idea:

- Let clients indicate their preferences.
- Each client may choose more that one person.
- Knowing all the preferences, arrange matching in such a way to maximize the number of couples.

Matrimonial agency problem

Question

How to maximize profits?, by matching as many couples as possible?

Idea:

- Let clients indicate their preferences.
- Each client may choose more that one person.
- Knowing all the preferences, arrange matching in such a way to maximize the number of couples.

Matrimonial agency problem

Question

How to maximize profits, by matching as many couples as possible?

Idea:

- Let clients indicate their preferences.
- Each client may choose more that one person.
- Knowing all the preferences, arrange matching in such a way to maximize the number of couples.

Matrimonial agency problem

Question

How to maximize profits, by matching as many couples as possible?

Idea:

- Let clients indicate their preferences.
- Each client may choose more that one person.
- Knowing all the preferences, arrange matching in such a way to maximize the number of couples.

Matrimonial agency problem

Question

How to maximize profits, by matching as many couples as possible?

Idea:

- Let clients indicate their preferences.
- Each client may choose more that one person.
- Knowing all the preferences, arrange matching in such a way to maximize the number of couples.

Matrimonial agency problem

What conditions should be satisfied for everyone to be matched?

Matrimonial agency problem

	Adam	Bob	Chris	David	Ed
Alice		x		x	x
Barbie		x	x		x
Celine			x		
Diana	x		x		
Elle	x		x		

Question

Is it possible to match 5 couples?

Main question

Question

What do these problems have in common??

Main question

Question

What do these problems have in common??

Graph theory!

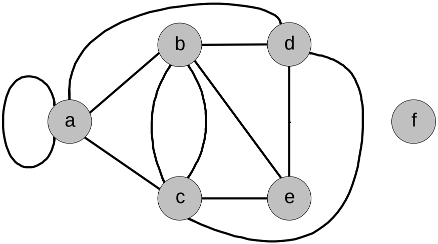
Outline

The most general definition

Definition

(Undirected) **multigraph (graph)** is a triple $G = (V, E, f)$, consisting of a **vertex set** V, an **edge set** E, and a mapping function $f\colon E \to \{\{u, v\}\colon u, v \in V\}$ which associates with each edge its **endpoints**.

Multigraph can have **loops** and **multiple edges**. We say that two vertices are **adjacent** if they are joined by some edge.

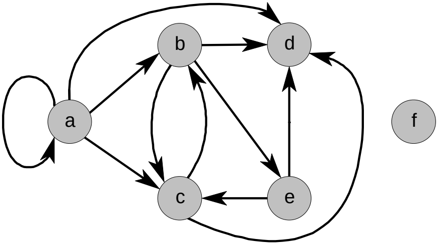


The most general definition

Definition

Directed multigraph is a triple $G = (V, A, f)$, consisting of a **vertex set** V, an **arc set** A (or directed edge set), and a **mapping function** $f\colon A \to V \times V$ which associates with each edge its **startpoint** and **endpoint**.

More about directed graphs you will hear later.

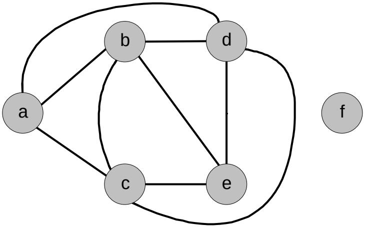


The most popular definition

Definition

Undirected graph (or *simple* undirected graph) is a pair $G = (V, E)$, consisting of a **vertex set** V, where an **edge set** $E \subseteq \{\{u, v\} : u, v \in V \wedge u \neq v\}$.

The above definition exclude an existence of loops or multiple edges. In simple graph an edge joining vertices u, v is denoted uv and $uv = vu$.



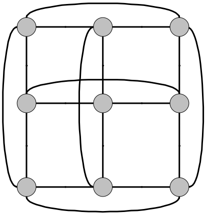
The most popular definition

Definition

Undirected graph (or *simple* undirected graph) is a pair $G = (V, E)$, consisting of a **vertex set** V, where an **edge set** $E \subseteq \{\{u, v\} : u, v \in V \wedge u \neq v\}$.

The above definition exclude an existence of loops or multiple edges.

In simple graph an edge joining vertices u, v is denoted uv and $uv = vu$.

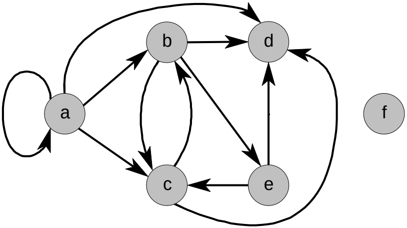


The most popular definition

Definition

Directed graph, (or **digraph** or simple directed graph) is a pair $G = (V, A)$, consisting of a **vertex set** V, where an **arc set** $A \subseteq V \times V$.

Digraph cannot have multiple arcs but can have directed loops.
In a digraph an arc joining vertices u, v is denoted uv, but $uv \neq vu$ for $u \neq v$.



Outline

Vertex degree

Definition

Degree of vertex $v \in V$ in multigraph $G = (V, E, f)$ is a number of its incident edges (with loops taken twice):

$$deg(v) = |\{e \in E \colon v \in f(e)\}| + |\{e \in E \colon v = f(e)\}|.$$

We denote $\delta(G) = \min\{\deg(v) \colon v \in V\}$ and $\Delta(G) = \max\{deg(v) \colon v \in V\}$.

Lemma (Handshake lemma)

In any multigraph G one has:

$$\sum_{v \in V(G)} \deg(v) = 2|E(G)|.$$

Subgraph

Definition

Let $G = (V, E)$ ba simple graph, $V_0 \subseteq V$ and $E_0 \subseteq E$. If pair (V_0, E_0) is a graph this graph $G_0 = (V_0, E_0)$ is called **subgraph** of graph G and denoted $G_0 \subseteq G$.

TODO: picture :-).

Induced subgraph

Definition

Let $G = (V, E)$ be a simple graph, $V_1 \subseteq V$ and $E_2 \subseteq E$. Let $E_1 = \left\{ e \in E \colon e \subseteq V_1 \right\}$. Graph $G_1 = (V_1, E_1)$ is called a **subgraph induced** by vertex set V_1, and denoted $G_1 = G[V_1]$. Let now $V_2 = \bigcup_{e \in E_2} e$. Graph $G_2 = (V_2, E_2)$ is a subgraph induced by edge set E_2.

TODO: picture :-).

Important are subgraphs obtained by vertex or edge removal. If $v \in V(G)$, $e \in E(G)$, then $G \setminus v = G[V(G) \setminus \{v\}]$ and $G - e = (V(G), E(G) \setminus \{e\})$.

Induced subgraph

Definition

Let $G = (V, E)$ be a simple graph, $V_1 \subseteq V$ and $E_2 \subseteq E$. Let $E_1 = \left\{e \in E\colon\ e \subseteq V_1\right\}$. Graph $G_1 = (V_1, E_1)$ is called a **subgraph induced** by vertex set V_1, and denoted $G_1 = G[V_1]$. Let now $V_2 = \bigcup_{e \in E_2} e$. Graph $G_2 = (V_2, E_2)$ is a subgraph induced by edge set E_2.

TODO: picture :-).

Important are subgraphs obtained by vertex or edge removal. If $v \in V(G)$, $e \in E(G)$, then $G \setminus v = G[V(G) \setminus \{v\}]$ and $G - e = (V(G), E(G) \setminus \{e\})$.

Induced subgraph

Definition

Let $G = (V, E)$ be a simple graph, $V_1 \subseteq V$ and $E_2 \subseteq E$. Let $E_1 = \left\{ e \in E \colon e \subseteq V_1 \right\}$. Graph $G_1 = (V_1, E_1)$ is called a **subgraph induced** by vertex set V_1, and denoted $G_1 = G[V_1]$. Let now $V_2 = \bigcup_{e \in E_2} e$. Graph $G_2 = (V_2, E_2)$ is a subgraph induced by edge set E_2.

TODO: picture :-).

Important are subgraphs obtained by vertex or edge removal. If $v \in V(G)$, $e \in E(G)$, then $G \setminus v = G[V(G) \setminus \{v\}]$ and $G - e = (V(G), E(G) \setminus \{e\})$.

Graph complement

Definition

Let $G = (V, E)$ be a simple graph. **Complement** of graph G is a simple graph $\overline{G} = (V, E')$, where

$$E' = \left\{ \{u, v\} \colon u, v \in V \wedge u \neq v \wedge \{u, v\} \notin E \right\}.$$

TODO: picture :-).

Complete graph and empty graph

Definition

Complete graph is a simple graph with with vertices pairwise adjacent. Complete graph on n vertices is denoted K_n.
If subset $W \subseteq V$ of graph $G = (V, E)$ induces complete graph we call it **clique**.

TODO: picture :-).

Definition

Empty graph is a graph with empty edge set. Empty graph on n vertices is denoted $\overline{K}_n$.
If subset $W \subseteq V$ of graph $G = (V, E)$ induces empty graph we call it **independent set**.

Complete graph and empty graph

Definition

Complete graph is a simple graph with with vertices pairwise adjacent. Complete graph on n vertices is denoted K_n.
If subset $W \subseteq V$ of graph $G = (V, E)$ induces complete graph we call it **clique**.

TODO: picture :-).

Definition

Empty graph is a graph with empty edge set. Empty graph on n vertices is denoted $\overline{K}_n$.
If subset $W \subseteq V$ of graph $G = (V, E)$ induces empty graph we call it **independent set**.

Bipartite graphs

Definition

Graph G is called **bipartite graph** if there exists a vertex set partition $V(G) = V_1 \cup V_2$, such that V_1 and V_2 are independent sets (recall: subgraphs $G[V_1]$ and $G[V_2]$ have no edges).

Definition

Simple graph $G = (V, E)$, with $V = V_1 \cup V_2$ for nonempty and disjoint sets V_1, V_2 is called **complete bipartite graph**, if

$$E = \Big\{\{u, v\} : u \in V_1 \wedge v \in V_2\Big\}.$$

When $|V_1| = m$ and $|V_2| = n$, such a complete bipartite graph is denoted $K_{m,n}$.

Bipartite graphs

Definition

Graph G is called **bipartite graph** if there exists a vertex set partition $V(G) = V_1 \cup V_2$, such that V_1 and V_2 are independent sets (recall: subgraphs $G[V_1]$ and $G[V_2]$ have no edges).

Definition

Simple graph $G = (V, E)$, with $V = V_1 \cup V_2$ for nonempty and disjoint sets V_1, V_2 is called **complete bipartite graph**, if

$$E = \Big\{\{u, v\} : u \in V_1 \wedge v \in V_2\Big\}.$$

When $|V_1| = m$ and $|V_2| = n$, such a complete bipartite graph is denoted $K_{m,n}$.

Graph traversals

Definition

A **walk** in multigraph G is any sequence $v_0, e_0, v_1, e_1, v_2, \ldots e_{k-1}, v_k$, $k \in \mathbb{N}$ such that for any i, $0 < i < k$ vertices v_i, v_{i+1} are joined by an edge e_i. A walk is called **trail** if it has no repeating edges. If, moreover, it has no repeating vertices it is called a **path**.

Graph traversals

Definition

A **walk** in multigraph G is any sequence $v_0, e_0, v_1, e_1, v_2, \dots e_{k-1}, v_k$, $k \in \mathbb{N}$ such that for any i, $0 < i < k$ vertices v_i, v_{i+1} are joined by an edge e_i. A walk is called **trail** if it has no repeating edges. If, moreover, it has no repeating vertices it is called a **path**.

Remark

The length of a walk, trail or path ...

Graph traversals

Definition

A **walk** in multigraph G is any sequence $v_0, e_0, v_1, e_1, v_2, \dots e_{k-1}, v_k$, $k \in \mathbb{N}$ such that for any i, $0 < i < k$ vertices v_i, v_{i+1} are joined by an edge e_i. A walk is called **trail** if it has no repeating edges. If, moreover, it has no repeating vertices it is called a **path**.

Remark

In a simple graphs endpoint vertices determine an edge — one can define a walk as a vertex sequence $v_0, v_1, v_2, \dots, v_k$. Subsequent vertices are adjacent, of course.

A path — ambiguity

Definition

A term **path** is also used to define a simple graph whose all vertices lies on a path (in a traversal sense). Path graph on n vertices is denoted P_n.

Closed trails and cycles

Definition

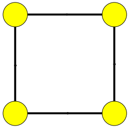
Closed trail $v_0, e_0, v_1, e_1, \ldots, v_k$ in multigraph (or graph) G ($v_0 = v_k$) is called **circuit** (or tour). A circuit is called a **cycle** if it has no repeating vertices, except $v_0 = v_k$.

Remark

Graph without cycles is called **acyclic**.

Remark

Analogously to the previous slide we define a cycle (in a graph sense). Cycle on n vertices is denoted C_n.



Closed trails and cycles

Definition

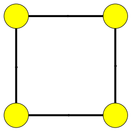
Closed trail $v_0, e_0, v_1, e_1, \ldots, v_k$ in multigraph (or graph) G ($v_0 = v_k$) is called **circuit** (or tour). A circuit is called a **cycle** if it has no repeating vertices, except $v_0 = v_k$.

Remark

Graph without cycles is called **acyclic**.

Remark

Analogously to the previous slide we define a cycle (in a graph sense). Cycle on n vertices is denoted C_n.



Graph connectivity

Definition

Vertices $u, v \in V$ of undirected graph G are **connected** if there exists uv-path in G, that means a path $v_0, \ldots, v_k$ such that $v_0 = u, v_k = v$. A graph with pairwise connected vertices is called **connected graph.**

Definition

Each maximal (thus induced) connected subgraph of graph G is called its **connected component.**

Graph connectivity

Definition

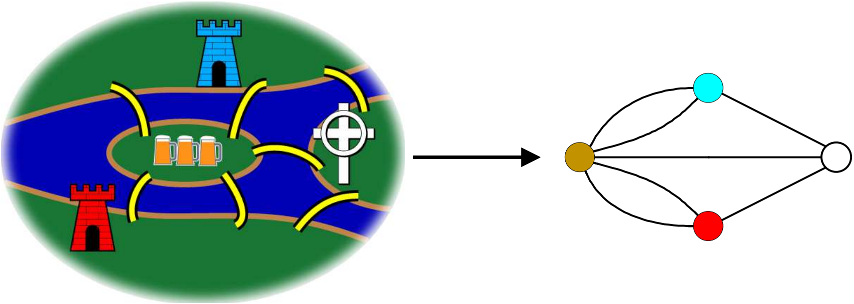
Vertices $u, v \in V$ of undirected graph G are **connected** if there exists uv-path in G, that means a path $v_0, \ldots, v_k$ such that $v_0 = u, v_k = v$. A graph with pairwise connected vertices is called **connected graph.**

Definition

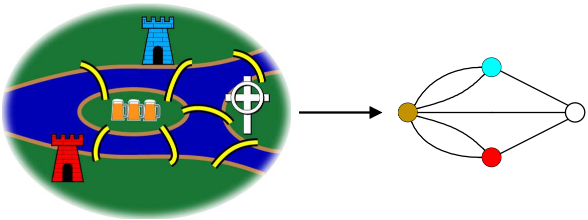
Each maximal (thus induced) connected subgraph of graph G is called its **connected component.**

Outline

The Königsberg Bridges Problem



The Königsberg Bridges Problem



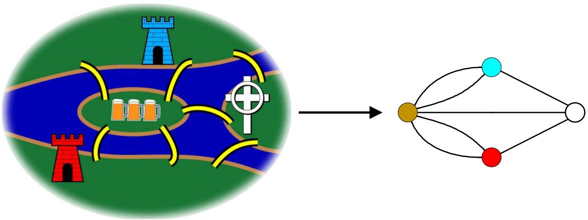
Theorem (Euler, 1736)

Finite and connected multigraph has Eulerian tour if and only if each its vertex has even degree.

Corollary

Finite and connected multigraph with exactly two odd degree vertices has Eulerian trail connecting this two vertices.

The Königsberg Bridges Problem



Theorem (Euler, 1736)

Finite and connected multigraph has Eulerian tour if and only if each its vertex has even degree.

Corollary

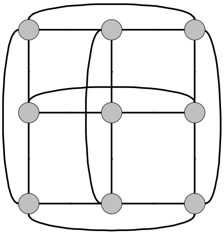
Finite and connected multigraph with exactly two odd degree vertices has Eulerian trail connecting this two vertices.

Outline

Planar graphs

Definition (Unformal)

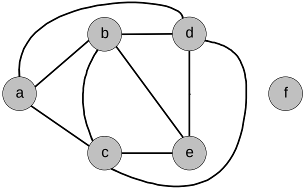
A **planar graph** is a graph that can be drawn on a plane without crossing its edges.



Planar graphs

Definition (Unformal)

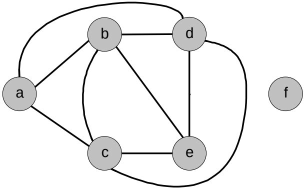
A **planar graph** is a graph that can be drawn on a plane without crossing its edges.



Planar graphs

Definition (Unformal)

A **planar graph** is a graph that can be drawn on a plane without crossing its edges.



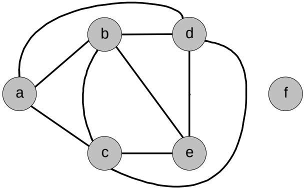
Question

Is K_5 planar?

Planar graphs

Definition (Unformal)

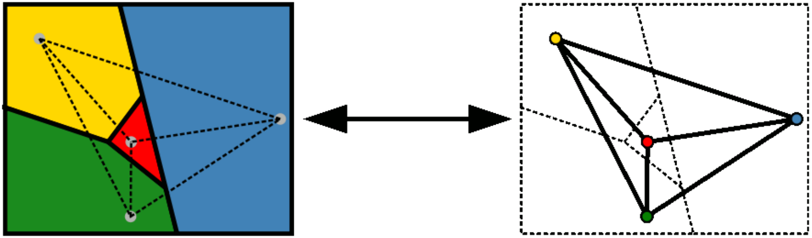
A **planar graph** is a graph that can be drawn on a plane without crossing its edges.



Question

Is K_5 planar? What about K_4?

What does it have to do with maps?



Not only planar graphs are worth to be coloured

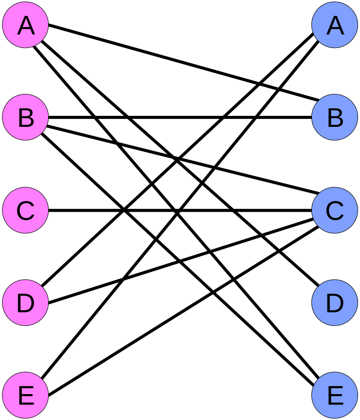
Example

A schedule (timetable) for teachers and students.

Outline

Marriage matchmaking

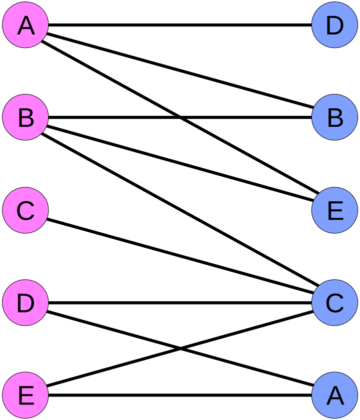
	Adam	Bob	Chris	David	Ed
Alice		x		x	x
Barbie		x	x		x
Celine			x		
Diana	x		x		
Elle	x		x		



Bipartite graph

Marriage matchmaking

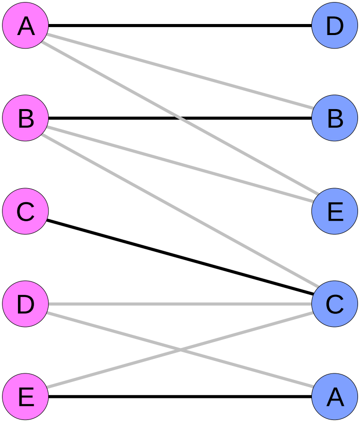
	Adam	Bob	Chris	David	Ed
Alice		x		x	x
Barbie		x	x		x
Celine			x		
Diana	x		x		
Elle	x		x		



Bipartite graph

Marriage matchmaking

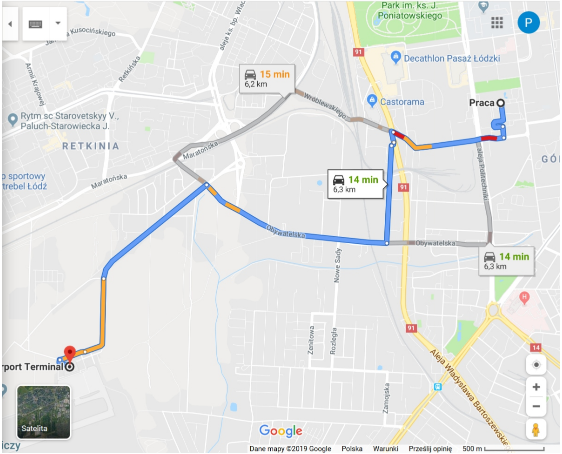
	Adam	Bob	Chris	David	Ed
Alice		x		x	x
Barbie		x	x		x
Celine			x		
Diana	x		x		
Elle	x		x		



Maximum matching

Classic algorithmic problems

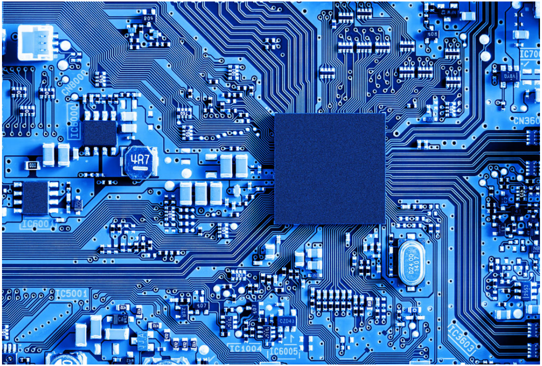
Eg. shortest paths, min spanning trees, max flows, etc.



Logistic and engineering

Supply line optimization

Circuit design



Data analysis & machine learning

Internal core of many methods and concepts in DA & ML (eg. decision trees, manifold learning, neural networks)

Social network modelling and analysis