Definition: Graph

Definition (Undirected Graph)

- Let \( V(G) = \{v_1, \ldots, v_n\} \) be a non-empty set of nodes and
- \( E(G) \) be a set or multiset of pairs from \( V(G) \) (set of edges).
- The sets \( V(G) \) and \( E(G) \) define the graph \( G = (V(G), E(G)) \).
- If \( G \) is uniquely determined, then we just write: \( V \) and \( E \).
- Or in other words \( G = (V, E) \).
- We always use as default writing: \( n = |V| \) and \( m = |E| \).
Definition: Graph

**Definition (Undirected Graph)**

- Let $V(G) = \{v_1, ..., v_n\}$ be a non-empty set of nodes and
- $E(G)$ be a set or multiset of pairs from $V(G)$ (set of edges).
- The sets $V(G)$ and $E(G)$ define the graph $G = (V(G), E(G))$.
- If $G$ is uniquely determined, then we just write: $V$ and $E$.
- Or in other words $G = (V, E)$.
- We always use as default writing: $n = |V|$ and $m = |E|$.
Definition: Graph

**Definition (Undirected Graph)**

- Let $V(G) = \{v_1, ..., v_n\}$ be a non-empty set of nodes and
- $E(G)$ be a set or multiset of pairs from $V(G)$ (set of edges).
- The sets $V(G)$ and $E(G)$ define the graph $G = (V(G), E(G))$.
- If $G$ is uniquely determined, then we just write: $V$ and $E$.
- Or in other words $G = (V, E)$.
- We always use as default writing: $n = |V|$ and $m = |E|$.
Definition: Graph

Definition (Undirected Graph)

- Let $V(G) = \{v_1, ..., v_n\}$ be a non-empty set of nodes and
- $E(G)$ be a set or multiset of pairs from $V(G)$ (set of edges).
- The sets $V(G)$ and $E(G)$ define the graph $G = (V(G), E(G))$.
- If $G$ is uniquely determined, then we just write: $V$ and $E$.
- Or in other words $G = (V, E)$.
- We always use as default writing: $n = |V|$ and $m = |E|$.
Definition: Graph

**Definition (Undirected Graph)**

- Let $V(G) = \{v_1, ..., v_n\}$ be a non-empty set of nodes and $E(G)$ be a set or multiset of pairs from $V(G)$ (set of edges).
- The sets $V(G)$ and $E(G)$ define the graph $G = (V(G), E(G))$.
- If $G$ is uniquely determined, then we just write: $V$ and $E$.
- Or in other words $G = (V, E)$.
- We always use as default writing: $n = |V|$ and $m = |E|$. 
Definition: Graph

**Definition (Undirected Graph)**

- Let $V(G) = \{v_1, ..., v_n\}$ be a non-empty set of nodes and
- $E(G)$ be a set or multiset of pairs from $V(G)$ (set of edges).
- The sets $V(G)$ and $E(G)$ define the graph $G = (V(G), E(G))$.
- If $G$ is uniquely determined, then we just write: $V$ and $E$.
- Or in other words $G = (V, E)$.
- *We always use as default writing: $n = |V|$ and $m = |E|$.*
Definition: Graph

Definition (Undirected Graph)

- Let $V(G) = \{v_1, \ldots, v_n\}$ be a non-empty set of nodes and
- $E(G)$ be a set or multiset of pairs from $V(G)$ (set of edges).
- The sets $V(G)$ and $E(G)$ define the graph $G = (V(G), E(G))$.
- If $G$ is uniquely determined, then we just write: $V$ and $E$.
- Or in other words $G = (V, E)$.
- We always use as default writing: $n = |V|$ and $m = |E|$.
Way of Speaking for Graphs

Definition (Way of Speaking)

- Let \( G = (V(G), E(G)) \) and \( e = (v, w) \in E(G) \).
- The nodes \( v, w \) are called connected (adjacent) by an edge \( e \).
- An edge \( e \) is called loop, if \( v = w \) holds.
- Two edges are called parallel, if they are the same.
- A graph without parallel edges is called simple.

As long as we do not state differently we will use in the following simple graph without loops.
Definition (Way of Speaking)

- Let $G = (V(G), E(G))$ and $e = (v, w) \in E(G)$.
- The nodes $v, w$ are called connected (adjacent) by an edge $e$.
- An edge $e$ is called loop, if $v = w$ holds.
- Two edges are called parallel, if they are the same.
- A graph without parallel edges is called simple.

As long as we do not state differently we will use in the following simple graph without loops.
**Way of Speaking for Graphs**

**Definition (Way of Speaking)**

- Let $G = (V(G), E(G))$ and $e = (v, w) \in E(G)$.
- The nodes $v, w$ are called **connected** (adjacent) by an edge $e$.
- An edge $e$ is called **loop**, if $v = w$ holds.
- Two edges are called **parallel**, if they are the same.
- A graph without parallel edges is called **simple**.

- As long as we do not state differently we will use in the following simple graph without loops.
Way of Speaking for Graphs

Definition (Way of Speaking)

- Let $G = (V(G), E(G))$ and $e = (v, w) \in E(G)$.
- The nodes $v, w$ are called connected (adjacent) by an edge $e$.
- An edge $e$ is called loop, if $v = w$ holds.
- Two edges are called parallel, if they are the same.
- A graph without parallel edges is called simple.

As long as we do not state differently we will use in the following simple graph without loops.
Way of Speaking for Graphs

Definition (Way of Speaking)

- Let $G = (V(G), E(G))$ and $e = (v, w) \in E(G)$.
- The nodes $v, w$ are called **connected** (adjacent) by an edge $e$.
- An edge $e$ is called **loop**, if $v = w$ holds.
- Two edges are called **parallel**, if they are the same.
- A graph without parallel edges is called **simple**.

- As long as we do not state differently we will use in the following simple graph without loops.
**Way of Speaking for Graphs**

**Definition (Way of Speaking)**

- Let $G = (V(G), E(G))$ and $e = (v, w) \in E(G)$.
- The nodes $v, w$ are called connected (adjacent) by an edge $e$.
- An edge $e$ is called loop, if $v = w$ holds.
- Two edges are called parallel, if they are the same.
- A graph without parallel edges is called simple.

- As long as we do not state differently we will use in the following simple graph without loops.
**Degree of a Node**

**Definition (Degree of a Node)**

- Let \( v \in V(G) \).
- With
  \[
  \text{deg}(v) = |\{e \in E(G) \mid e = (v, v'), v' \in V(G) \setminus \{v\}\}| 
  \]
  we denote the degree of a Node (degree) of \( v \).

- \( \text{deg}(v_0) = 4 \).
- \( \text{deg}(v_1) = 3 \).
- \( \text{deg}(v_4) = 6 \).
- \( \text{deg}(v_5) = 6 \).
**Degree of a Node**

**Definition (Degree of a Node)**

- Let $v \in V(G)$.
- With
  
  $$\deg(v) = |\{e \in E(G) \mid e = (v, v'), v' \in V(G) \setminus \{v\}\}|$$

we denote the degree of a Node (degree) of $v$.

- $\deg(v_0) = 4$.
- $\deg(v_1) = 3$.
- $\deg(v_4) = 6$.
- $\deg(v_5) = 6$. 
Degree of a Node

Definition (Degree of a Node)

- Let \( v \in V(G) \).
- With
  \[
  \text{deg}(v) = |\{e \in E(G) \mid e = (v, v'), v' \in V(G) \setminus \{v\}\}| 
  \]
  we denote the degree of a Node (degree) of \( v \).

- \( \text{deg}(v_0) = 4 \).
- \( \text{deg}(v_1) = 3 \).
- \( \text{deg}(v_4) = 6 \).
- \( \text{deg}(v_5) = 6 \).
**Degree of a Node**

**Definition (Degree of a Node)**

- Let \( v \in V(G) \).
- With
  \[
  \deg(v) = |\{e \in E(G) \mid e = (v, v'), v' \in V(G) \setminus \{v\}\}|
  \]
  we denote the degree of a Node (degree) of \( v \).

- \( \deg(v_0) = 4 \).
- \( \deg(v_1) = 3 \).
- \( \deg(v_4) = 6 \).
- \( \deg(v_5) = 6 \).
Degree of a Node

**Definition (Degree of a Node)**

- Let $v \in V(G)$.
- With
  \[ \text{deg}(v) = |\{ e \in E(G) \mid e = (v, v'), v' \in V(G) \setminus \{v\} \}| \]
  we denote the **degree of a Node** (**degree**) of $v$.

- $\text{deg}(v_0) = 4$.
- $\text{deg}(v_1) = 3$.
- $\text{deg}(v_4) = 6$.
- $\text{deg}(v_5) = 6$.
Degree of a Node

Definition (Degree of a Node)

Let $v \in V(G)$.

With

$$\text{deg}(v) = |\{e \in E(G) \mid e = (v, v'), v' \in V(G) \setminus \{v\}\}|$$

we denote the degree of a Node (degree) of $v$.

- $\text{deg}(v_0) = 4$.
- $\text{deg}(v_1) = 3$.
- $\text{deg}(v_4) = 6$.
- $\text{deg}(v_5) = 6$. 
Degree of a Node

**Definition (Degree of a Node)**

- Let \( v \in V(G) \).
- With

\[
\deg(v) = \left| \{e \in E(G) \mid e = (v, v'), v' \in V(G) \setminus \{v\} \} \right|
\]

we denote the **degree of a Node (degree)** of \( v \).

- \( \deg(v_0) = 4 \).
- \( \deg(v_1) = 3 \).
- \( \deg(v_4) = 6 \).
- \( \deg(v_5) = 6 \).
Degree of a Node

**Definition (Degree of a Node)**

- Let $v \in V(G)$.
- With

$$\deg(v) = |\{e \in E(G) \mid e = (v, v'), v' \in V(G) \setminus \{v\}\}|$$

we denote the **degree of a Node (degree)** of $v$.

- $\deg(v_0) = 4$.
- $\deg(v_1) = 3$.
- $\deg(v_4) = 6$.
- $\deg(v_5) = 6$. 

**Diagram:**

![Graph Diagram]
Handshake Theorem

Theorem

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

Proof: Each edge connects two nodes.

Theorem

The number of nodes of odd degree is even.

Proof:

\[ \sum_{v \in V(G)} \deg(v) + \sum_{v \in V(G)} \deg(v) \mod 2 = 0 \]

\[ \sum_{v \in V(G)} \deg(v) \mod 2 = 1 \]

\[ \sum_{v \in V(G)} \deg(v) = 2|E(G)| \]
Handshake Theorem

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

Proof: Each edge connects two nodes.

The number of nodes of odd degree is even.

Proof:

\[ \sum_{v \in V(G)} \deg(v) + \sum_{v \in V(G)} \deg(v) = 2|E(G)| \]

\[ \deg(v) \mod 2 = 0 \]

\[ \deg(v) \mod 2 = 1 \]
Handshake Theorem

Theorem

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

Proof: Each edge connects two nodes.

Theorem

*The number of nodes of odd degree is even.*

Proof:

\[ \sum_{v \in V(G)} \deg(v) + \sum_{v \in V(G)} \deg(v) \mod 2 = 0 \]

\[ \sum_{v \in V(G)} \deg(v) \mod 2 = 1 \]
Handshake Theorem

Theorem

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

Proof: Each edge connects two nodes.

Theorem

The number of nodes of odd degree is even.

Proof:

\[ \sum_{\deg(v) \mod 2 = 0} \deg(v) + \sum_{\deg(v) \mod 2 = 1} \deg(v) = 2|E(G)| \]
Regular and Complete

Definition (Regular)
A graph $G$ is called $k$-regular, iff for all $v \in V(G)$ we have: $d(v) = k$.

Definition (Complete)
A graph $G$ is called complete, iff all pairs of nodes $a, b$ from $V$ holds: $(a, b) \in E$.

Notation: $K_n$. 
Regular and Complete

**Definition (Regular)**

A graph $G$ is called $k$-regular, iff for all $v \in V(G)$ we have: $d(v) = k$.

**Definition (Complete)**

A graph $G$ is called complete, iff all pairs of nodes $a, b$ from $V$ holds: $(a, b) \in E$.

Notation: $K_n$. 
**Definition (Regular)**

A graph $G$ is called $k$-regular, iff for all $v \in V(G)$ we have: $d(v) = k$.

---

**Definition (Complete)**

A graph $G$ is called complete, iff all pairs of nodes $a, b$ from $V$ holds: $(a, b) \in E$.

- Notation: $K_n$. 
**Definition (Bipartite)**

A Graph $G$ is called **bipartite**, iff $V$ may be split into disjoint set $V'$, $V''$, such that each edge connects only nodes from both partitions.

- **Notation:** $G = (V', V'', E)$

**Definition (Complete bipartite)**

A Graph $G$ is called **complete bipartite**, iff $V$ may be split into disjoint set $V'$, $V''$, and $E = \{(a, b) \mid a \in V', \ b \in V''\}$.

- **Notation:** $K_{p,q}$ with $p = |V'|$ and $q = |V''|$.
- **Star,** iff $S_n = K_{1,n-1}$. 
Special Graphs

Definition (Bipartite)

A Graph $G$ is called bipartite, iff $V$ may be split into disjoint set $V'$, $V''$, such that each edge connects only nodes from both partitions.

- Notation: $G = (V', V'', E)$

Definition (Complete Bipartite)

A Graph $G$ is called complete bipartite, iff $V$ may be split into disjoint set $V'$, $V''$, and $E = \{(a, b) \mid a \in V', b \in V''\}$.

- Notation: $K_{p,q}$ with $p = |V'|$ and $q = |V''|$.
- Star, iff $S_n = K_{1,n−1}$. 

Special Graphs

**Definition (Bipartite)**

A Graph $G$ is called **bipartite**, iff $V$ may be split into disjoint set $V'$, $V''$, such that each edge connects only nodes from both partitions.

- Notation: $G = (V', V'', E)$

**Definition (Complete bipartite)**

A Graph $G$ is called **complete bipartite**, iff $V$ may be split into disjoint set $V'$, $V''$, and

$$E = \{(a, b) \mid a \in V', \ b \in V''\}.$$

- Notation: $K_{p,q}$ with $p = |V'|$ and $q = |V''|$.
- **Star**, iff $S_n = K_{1,n-1}$.  

Examples
Examples
Examples
Definition (Subgraph)

A Graph $H = (V(H), E(H))$ is a subgraph of $G = (V(G), E(G))$, iff $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. 
Subgraphs

Definition (Subgraph)

A Graph $H = (V(H), E(H))$ is called a subgraph of $G = (V(G), E(G))$, iff $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$.
Subgraphs

Definition (Subgraph)

- A Graph $H = (V(H), E(H))$ is call a subgraph of $G = (V(G), E(G))$,
- iff $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$.
Subgraphs

**Definition (node-induced subgraph)**

- A graph $H = (V(H), E(H))$ is a node-induced subgraph of $G = (V(G), E(G))$,

- iff $V(H) \subseteq V(G)$ and $E(H) = \{(a, b) \in E(G) \mid a, b \in V(H)\}$.
Subgraphs

**Definition (node-induced subgraph)**

- A graph $H = (V(H), E(H))$ is a node-induced subgraph of $G = (V(G), E(G))$, if $V(H) \subseteq V(G)$ and $E(H) = \{(a, b) \in E(G) \mid a, b \in V(H)\}$.
**Definition (node-induced subgraph)**

- A graph \( H = (V(H), E(H)) \) is a node-induced subgraph of \( G = (V(G), E(G)) \),

- iff \( V(H) \subseteq V(G) \) and \( E(H) = \{ (a, b) \in E(G) \mid a, b \in V(H) \} \).
**Isomorph Graphs**

**Definition (Isomorph)**

- Two graphs $G$ and $H$ are called isomorph,
- iff there is a bijective mapping $f : V(G) \leftrightarrow V(H)$,
- such that for all $v, w \in V(G)$ hold:
- $(v, w) \in E(G)$, iff $(f(v), f(w)) \in E(H)$. 

![Graphs](attachment:Graphs.png)
Isomorph Graphs

Definition (Isomorph)

- Two graphs $G$ and $H$ are called isomorph,
- iff there is a bijective mapping $f : V(G) \mapsto V(H)$,
- such that for all $v, w \in V(G)$ hold:
- $(v, w) \in E(G), \text{ iff } (f(v), f(w)) \in E(H)$. 

\[ \begin{align*}
&v_0 \rightarrow v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow v_4 \rightarrow v_5 \\
&a_0 \rightarrow a_1 \rightarrow a_2 \\
&b_0 \rightarrow b_1 \rightarrow b_2
\end{align*} \]
Isomorph Graphs

**Definition (Isomorph)**

Two graphs $G$ and $H$ are called *isomorph*,
iff there is a bijective mapping $f : V(G) \leftrightarrow V(H),$ such that for all $v, w \in V(G)$ hold:

$(v, w) \in E(G),$ iff $(f(v), f(w)) \in E(H).$
**Isomorph Graphs**

**Definition (Isomorph)**

- Two graphs $G$ and $H$ are called isomorph,
- iff there is a bijective mapping $f : V(G) \rightarrow V(H)$,
- such that for all $v, w \in V(G)$ hold:
- $(v, w) \in E(G)$, iff $(f(v), f(w)) \in E(H)$.
Isomorph Graphs

Definition (Isomorph)

- Two graphs $G$ and $H$ are called isomorph,
- iff there is a bijective mapping $f : V(G) \mapsto V(H),$
- such that for all $v, w \in V(G)$ hold:
- $(v, w) \in E(G), \text{ iff } (f(v), f(w)) \in E(H).$
Connectivity

Definition

A graph \( G = (V, E) \) is called connected, iff between any two different nodes \( a, b \) exists a path from \( a \) to \( b \).
Connectivity

**Definition**

A graph $G = (V, E)$ is called connected, iff between any two different nodes $a, b$ exists a path from $a$ to $b$. 

![Graph Diagram]
A graph $G = (V, E)$ is called connected, iff between any two different nodes $a, b$ exists a path from $a$ to $b$. 
A graph $G = (V, E)$ is called connected, iff between any two different nodes $a, b$ exists a path from $a$ to $b$. 

![Graph Diagram]
**Definition**

A graph $G = (V, E)$ is called connected, iff between any two different nodes $a, b$ exists a path from $a$ to $b$. 
A graph \( G = (V, E) \) is called connected, iff between any two different nodes \( a, b \) exists a path from \( a \) to \( b \).
Node-Separator

**Definition**

Let $G = (V, E)$, $V' \subset V$ is called a node-separator (vertex cut), iff $G - V'$ is not connected.

**Notation:** $G - V' := (V \setminus V', \{(a, b) \in E \mid a, b \in V \setminus V'\})$

**Definition**

If $\{v\}$ is a node-separator, then $v$ is called articulation point.

**Theorem**

*Only cliques $K_n$ do not have any node-separator.*
Node-Separator

Definition

Let $G = (V, E)$, $V' \subset V$ is called a node-separator (vertex cut), iff $G - V'$ is not connected.

Notation: $G - V' := (V \setminus V', \{(a, b) \in E \mid a, b \in V \setminus V'\})$

Definition

If $\{v\}$ is a node-separator, then $v$ is called articulation point.

Theorem

Only cliques $K_n$ do not have any node-separator.
Node-Separator

**Definition**

Let \( G = (V, E) \), \( V' \subset V \) is called a node-separator (vertex cut), iff \( G - V' \) is not connected.

Notation: \( G - V' := (V \setminus V', \{(a, b) \in E \mid a, b \in V \setminus V'\}) \)

**Definition**

If \( \{v\} \) is a node-separator, then \( v \) is called articulation point.

**Theorem**

*Only cliques \( K_n \) do not have any node-separator.*
Example
Example
Example
Example
Definition
Let $G = (V, E)$. $E' \subset E$ is called edge-separator (edge cut), iff $G - E'$ is not connected.

Notation: $G - E' := (V, E \setminus E')$

Definition
If $\{v, w\}$ is an edge-separator, then $\{v, w\}$ is called a bridge.

Theorem
An minimal edge-separator $E'$ of $G = (V, E)$ induces a 2-partite graph. Or in other words: $G = (V, E')$ is a 2-partite graph.
**Definition**

Let $G = (V, E)$. $E' \subset E$ is called edge-separator (edge cut), iff $G - E'$ is not connected.

**Notation:** $G - E' := (V, E \setminus E')$

**Definition**

If $\{v, w\}$ is an edge-separator, then $\{v, w\}$ is called a bridge.

**Theorem**

An minimal edge-separator $E'$ of $G = (V, E)$ induces a 2-partite graph. Or in other words: $G = (V, E')$ is a 2-partite graph.
**Edge-Separator**

**Definition**

Let $G = (V, E)$. $E' \subseteq E$ is called edge-separator (edge cut), iff $G - E'$ is not connected.

Notation: $G - E' := (V, E \setminus E')$

**Definition**

If $\{v, w\}$ is an edge-separator, then $\{v, w\}$ is called a bridge.

**Theorem**

An minimal edge-separator $E'$ of $G = (V, E)$ induces a 2-partite graph. Or in other words: $G = (V, E')$ is a 2-partite graph.
Example
Example
Connectivity

Definition

Let $G = (V, E)$ and $k$ minimal with: $\exists V' \subseteq V : |V'| = k$ and $G - V'$ is not connected or trivial. Then we call $G$ $k$-connected.

A $k$-connected Graph is also $k - 1$-connected.

Notation: $\kappa(G) = k$

Definition

Let $G = (V, E)$ and $k$ minimal with: $\exists E' \subseteq E : |E'| = k$ and $G - E'$ is not connected or trivial. Then we call $G$ $k$-edge-connected.

A $k$-edge-connected Graph is also $k - 1$-edge-connected.

Notation: $\lambda(G) = k$
**Connectivity**

**Definition**

Let $G = (V, E)$ and $k$ minimal with: $\exists V' \subset V : |V'| = k$ and $G - V'$ is not connected or trivial. Then we call $G k$-connected.

A $k$-connected Graph is also $k - 1$-connected.

**Notation:** $\kappa(G) = k$

**Definition**

Let $G = (V, E)$ and $k$ minimal with: $\exists E' \subset E : |E'| = k$ and $G - E'$ is not connected or trivial. Then we call $G k$-edge-connected.

A $k$-edge-connected Graph is also $k - 1$-edge-connected.

**Notation:** $\lambda(G) = k$
Statements on Connectivity

Theorem
For any graph $G = (V, E)$ we have:
\[ \kappa(G) \leq \lambda(G) \leq \delta(G) \]

Notation: $\delta(G) := \min\{\deg(v) \mid v \in V\}$

Theorem
For all integer numbers $0 < a \leq b \leq c$ there are graphs $G$ with:
\[ \kappa(G) = a, \ \lambda(G) = b, \ \delta(G) = c \]

Theorem
Let $G = (V, E)$ be a graph with: $|V| = n$ and $\delta(G) \geq n/2$. Then we have:
\[ \lambda(G) = \delta(G) \]
Statements on Connectivity

**Theorem**

*For any graph $G = (V, E)$ we have:*

$$\kappa(G) \leq \lambda(G) \leq \delta(G)$$

**Notation:** $\delta(G) := \min\{\deg(v) \mid v \in V\}$

**Theorem**

*For all integer numbers $0 < a \leq b \leq c$ there are graphs $G$ with:*

$$\kappa(G) = a, \quad \lambda(G) = b, \quad \delta(G) = c$$

**Theorem**

*Let $G = (V, E)$ be a graph with: $|V| = n$ and $\delta(G) \geq n/2$. Then we have:*

$$\lambda(G) = \delta(G)$$
### Statements on Connectivity

**Theorem**

For any graph $G = (V, E)$ we have:

$$
\kappa(G) \leq \lambda(G) \leq \delta(G)
$$

**Notation:** $\delta(G) := \min\{\deg(v) \mid v \in V\}$

### Theorem

For all integer numbers $0 < a \leq b \leq c$ there are graphs $G$ with:

$$
\kappa(G) = a, \quad \lambda(G) = b, \quad \delta(G) = c
$$

### Theorem

Let $G = (V, E)$ be a graph with: $|V| = n$ and $\delta(G) \geq n/2$. Then we have:

$$
\lambda(G) = \delta(G)
$$
Statements on Node-Connectivity

**Theorem**

Let $G = (V, E)$ with: $|V| = n$ and $|E| = e$. Then is the maximal connectivity (maximal $k$ with $G$ is $k$-connected) of $G$:

- $0$ falls if $e < n - 1$
- $2 \cdot \frac{e}{n}$ if $e \geq n - 1$

**Theorem**

Let $G = (V, E)$ connected. The following statements are equivalent:

1. $v \in V$ is a node-separator.
2. $\exists a, b \in V$: $a, b \neq v$: each path from $a$ to $b$ traverses via $v$.
3. $\exists A, B$: $A \cup B = V \setminus \{v\}$ and each path from $a \in A$ to $b \in B$ traverses via $v$. 
Statements on Node-Connectivity

**Theorem**

Let $G = (V, E)$ with: $|V| = n$ and $|E| = e$. Then is the maximal connectivity (maximal $k$ with $G$ is $k$-connected) of $G$:

- $0$ falls if $e < n - 1$
- $2 \cdot e/n$ if $e \geq n - 1$

**Theorem**

Let $G = (V, E)$ connected. The following statements are equivalent:

1. $v \in V$ is a node-separator.
2. $\exists a, b \in V: a, b \neq v$: each path from $a$ to $b$ traverses via $v$.
3. $\exists A, B: A \cup B = V \setminus \{v\}$ and each path from $a \in A$ to $b \in B$ traverses via $v$. 
Statements on Node-Connectivity

Theorem

Let $G = (V, E)$ with: $|V| = n$ and $|E| = e$. Then is the maximal connectivity (maximal $k$ with $G$ is $k$-connected) of $G$:

$$
\begin{align*}
0 & \text{ falls if } e < n - 1 \\
2 \cdot e/n & \text{ if } e \geq n - 1
\end{align*}
$$

Theorem

Let $G = (V, E)$ connected. The following statements are equivalent:

1. $v \in V$ is a node-separator.
2. $\exists a, b \in V: a, b \neq v$: each path from $a$ to $b$ traverses via $v$.
3. $\exists A, B: A \cup B = V \setminus \{v\}$ and each path from $a \in A$ to $b \in B$ traverses via $v$. 
Theorem

Let $G = (V, E)$ with: $|V| = n$ and $|E| = e$. Then is the maximal connectivity (maximal $k$ with $G$ is $k$-connected) of $G$:

$$
0 \text{ falls } e < n - 1,
$$
$$
2 \cdot \frac{e}{n} \text{ if } e \geq n - 1.
$$

Theorem

Let $G = (V, E)$ connected. The following statements are equivalent:

1. $v \in V$ is a node-separator.
2. $\exists a, b \in V: a, b \neq v$: each path from $a$ to $b$ traverses via $v$.
3. $\exists A, B: A \cup B = V \setminus \{v\}$ and each path from $a \in A$ to $b \in B$ traverses via $v$. 
Statements on Node-Connectivity

Theorem

Let $G = (V, E)$ with: $|V| = n$ and $|E| = e$. Then is the maximal connectivity (maximal $k$ with $G$ is $k$-connected) of $G$:

$$
\begin{align*}
0 & \text{ falls } e < n - 1 \\
2 \cdot \frac{e}{n} & \text{ if } e \geq n - 1
\end{align*}
$$

Theorem

Let $G = (V, E)$ connected. The following statements are equivalent:

1. $v \in V$ is a node-separator.
2. $\exists a, b \in V: a, b \neq v$: each path from $a$ to $b$ traverses via $v$.
3. $\exists A, B: A \cup B = V \setminus \{v\}$ and each path from $a \in A$ to $b \in B$ traverses via $v$. 
Let $G = (V, E)$ be connected. The following statements are equivalent:

1. $e \in E$ is a edge-separator.
2. $e$ is not in any simple cycle of $G$.
3. $\exists a, b \in E$: each path from $a$ to $b$ traverses via $e$.
4. $\exists A, B: A \cup B = V$ and each path from $a \in A$ to $b \in B$ traverses via $e$. 
Let $G = (V, E)$ be connected. The following statements are equivalent:

1. $e \in E$ is an edge-separator.
2. $e$ is not in any simple cycle of $G$.
3. $\exists a, b \in E$: each path from $a$ to $b$ traverses via $e$.
4. $\exists A, B: A \cup B = V$ and each path from $a \in A$ to $b \in B$ traverses via $e$. 

Theorem
Theorem

Let $G = (V, E)$ be connected. The following statements are equivalent:

1. $e \in E$ is a edge-separator.
2. $e$ is not in any simple cycle of $G$.
3. $\exists a, b \in E$: each path from $a$ to $b$ traverses via $e$.
4. $\exists A, B: A \cup B = V$ and each path from $a \in A$ to $b \in B$ traverses via $e$. 
Theorem

Let $G = (V, E)$ be connected. The following statements are equivalent:

1. $e \in E$ is a edge-separator.
2. $e$ is not in any simple cycle of $G$.
3. $\exists a, b \in E$: each path from $a$ to $b$ traverses via $e$.
4. $\exists A, B: A \cup B = V$ and each path from $a \in A$ to $b \in B$ traverses via $e$. 
Statements on Edge-Connectivity

**Theorem**

Let $G = (V, E)$ be connected. The following statements are equivalent:

1. $e \in E$ is a edge-separator.
2. $e$ is not in any simple cycle of $G$.
3. $\exists a, b \in E$: each path from $a$ to $b$ traverses via $e$.
4. $\exists A, B: A \cup B = V$ and each path from $a \in A$ to $b \in B$ traverses via $e$.
Definition

Let $G = (V, E)$ and $(a, b) = e \in E$. The subdivision of an edge $e$ results in graph $G = (V \cup \{v\}, E \cup \{(a, v), (v, b)\} \setminus \{e\})$.

Definition

A set of paths of $G = (V, E)$ is called intern-node-disjoint, iff no two paths share an internal-node. The internal nodes are all except the start and the end node.
**Definition**

Let $G = (V, E)$ and $(a, b) = e \in E$. The subdivision of an edge $e$ results in graph $G = (V \cup \{v\}, E \cup \{(a, v), (v, b)\} \setminus \{e\})$

**Definition**

A set of paths of $G = (V, E)$ is called intern-node-disjoint, iff no two paths share an internal-node. The internal nodes are all except the start and the end node.
Definition

Let $G = (V, E)$ and $(a, b) = e \in E$. The subdivision of an edge $e$ results in graph $G = (V \cup \{v\}, E \cup \{(a, v), (v, b)\} \setminus \{e\})$.

Definition

A set of paths of $G = (V, E)$ is called intern-node-disjoint, iff no two paths share an internal-node. The internal nodes are all except the start and the end node.
Theorem

Let $G = (V, E)$ with $|V| \geq 3$. The following statements are equivalent:

1. $G$ is 2-connected.
2. Each node pair is connected by two intern-node-disjoint paths.
3. Each node pair is on a common simple cycle.
4. There exits an edge and each node together with this edge is on a common simple cycle.
5. There exit two edges and each pair of edges is on a common simple cycle.
6. For each pair of nodes $a, b$ and an edge $e$ exists a simple path from $a$ to $b$ traversing $e$.
7. For three nodes $a, b, c$ exists a path from $a$ to $b$ traversing $c$.
8. For three nodes $a, b, c$ exists a path from $a$ to $b$ avoiding $c$. 
Theorem

Let $G = (V, E)$ with $|V| \geq 3$. The following statements are equivalent:

1. $G$ is 2-connected.

2. Each node pair is connected by two intern-node-disjoint paths.

3. Each node pair is on a common simple cycle.

4. There exits an edge and each node together with this edge is on a common simple cycle.

5. There exit two edges and each pair of edges is on a common simple cycle.

6. For each pair of nodes $a, b$ and an edge $e$ exists a simple path from $a$ to $b$ traversing $e$.

7. For three nodes $a, b, c$ exists a path from $a$ to $b$ traversing $c$.

8. For three nodes $a, b, c$ exists a path from $a$ to $b$ avoiding $c$. 
Let $G = (V, E)$ with $|V| \geq 3$. The following statements are equivalent:

1. $G$ is 2-connected.
2. Each node pair is connected by two intern-node-disjoint paths.
3. Each node pair is on a common simple cycle.
4. There exits an edge and each node together with this edge is on a common simple cycle.
5. There exit two edges and each pair of edges is on a common simple cycle.
6. For each pair of nodes $a, b$ and an edge $e$ exists a simple path from $a$ to $b$ traversing $e$.
7. For three nodes $a, b, c$ exists a path from $a$ to $b$ traversing $c$.
8. For three nodes $a, b, c$ exists a path from $a$ to $b$ avoiding $c$. 
Theorem

Let $G = (V, E)$ with $|V| \geq 3$. The following statements are equivalent:

1. $G$ is 2-connected.
2. Each node pair is connected by two intern-node-disjoint paths.
3. Each node pair is on a common simple cycle.
4. There exits an edge and each node together with this edge is on a common simple cycle.
5. There exit two edges and each pair of edges is on a common simple cycle.
6. For each pair of nodes $a, b$ and an edge $e$ exists a simple path from $a$ to $b$ traversing $e$.
7. For three nodes $a, b, c$ exists a path from $a$ to $b$ traversing $c$.
8. For three nodes $a, b, c$ exists a path from $a$ to $b$ avoiding $c$. 
Theorem

Let $G = (V, E)$ with $|V| \geq 3$. The following statements are equivalent:

1. $G$ is 2-connected.

2. Each node pair is connected by two intern-node-disjoint paths.

3. Each node pair is on a common simple cycle.

4. There exits an edge and each node together with this edge is on a common simple cycle.

5. There exit two edges and each pair of edges is on a common simple cycle.

6. For each pair of nodes $a, b$ and an edge $e$ exists a simple path from $a$ to $b$ traversing $e$.

7. For three nodes $a, b, c$ exists a path from $a$ to $b$ traversing $c$.

8. For three nodes $a, b, c$ exists a path from $a$ to $b$ avoiding $c$. 
Theorem

Let $G = (V, E)$ with $|V| \geq 3$. The following statements are equivalent:

1. $G$ is 2-connected.
2. Each node pair is connected by two intern-node-disjoint paths.
3. Each node pair is on a common simple cycle.
4. There exits an edge and each node together with this edge is on a common simple cycle.
5. There exit two edges and each pair of edges is on a common simple cycle.
6. For each pair of nodes $a, b$ and an edge $e$ exists a simple path from $a$ to $b$ traversing $e$.
7. For three nodes $a, b, c$ exists a path from $a$ to $b$ traversing $c$.
8. For three nodes $a, b, c$ exists a path from $a$ to $b$ avoiding $c$. 
Theorem

Let $G = (V, E)$ with $|V| \geq 3$. The following statements are equivalent:

1. $G$ is 2-connected.

2. Each node pair is connected by two intern-node-disjoint paths.

3. Each node pair is on a common simple cycle.

4. There exists an edge and each node together with this edge is on a common simple cycle.

5. There exist two edges and each pair of edges is on a common simple cycle.

6. For each pair of nodes $a, b$ and an edge $e$ exists a simple path from $a$ to $b$ traversing $e$.

7. For three nodes $a, b, c$ exists a path from $a$ to $b$ traversing $c$.

8. For three nodes $a, b, c$ exists a path from $a$ to $b$ avoiding $c$. 
Theorem

Let \( G = (V, E) \) with \( |V| \geq 3 \). The following statements are equivalent:

1. \( G \) is 2-connected.
2. Each node pair is connected by two intern-node-disjoint paths.
3. Each node pair is on a common simple cycle.
4. There exits an edge and each node together with this edge is on a common simple cycle.
5. There exit two edges and each pair of edges is on a common simple cycle.
6. For each pair of nodes \( a, b \) and an edge \( e \) exists a simple path from \( a \) to \( b \) traversing \( e \).
7. For three nodes \( a, b, c \) exists a path from \( a \) to \( b \) traversing \( c \).
8. For three nodes \( a, b, c \) exists a path from \( a \) to \( b \) avoiding \( c \).
Theorem

Let $G = (V, E)$ with $|V| \geq 3$. The following statements are equivalent:

1. $G$ is 2-connected.
2. Each node pair is connected by two intern-node-disjoint paths.
3. Each node pair is on a common simple cycle.
4. There exits an edge and each node together with this edge is on a common simple cycle.
5. There exit two edges and each pair of edges is on a common simple cycle.
6. For each pair of nodes $a, b$ and an edge $e$ exists a simple path from $a$ to $b$ traversing $e$.
7. For three nodes $a, b, c$ exists a path from $a$ to $b$ traversing $c$.
8. For three nodes $a, b, c$ exists a path from $a$ to $b$ avoiding $c$. 
Theorem

Let $G = (V, E)$ $k$-connected. Then any $k$ nodes are on a common simple cycle.

Notation: Let $(G = V, E)$ and $(H = W, F)$ graphs
$G + W = (V \cup W, E \cup F \cup \{(a, b) \mid a \in V, b \in W\})$

Theorem

A graph $G$ is 3-connected, iff $G$ may be constructed from the wheel $W_i = K_1 + C_i$ $(i \geq 4)$ by the following operations:

1. Adding a new edge.
2. Splitting a node of degree $\geq 4$ into two connected nodes of degree $\geq 3$. 
Basic Definitions
Connectivity of Graphs
Flows
Matchings
Factors of Graphs
Posets

Statements (1:22.2)

Theorem

Let $G = (V, E)$ $k$-connected. Then any $k$ nodes are on a common simple cycle.

Notation: Let $(G = V, E)$ and $(H = W, F)$ graphs
$G + W = (V \cup W, E \cup F \cup \{(a, b) \mid a \in V, b \in W\})$

Theorem

A graph $G$ is 3-connected, iff $G$ may be constructed from the wheel $W_i = K_1 + C_i$ ($i \geq 4$) by the following operations:

1. Adding a new edge.
2. Splitting a node of degree $\geq 4$ into two connected nodes of degree $\geq 3$. 
Statements on k-Connectivity

**Theorem (Menger’s Theorem)**

*G is k-connected, iff any two node are connected by k intern-node-disjoint paths.*

**Theorem (Menger's Theorem)**

*G is k-edge-connected, iff any two node are connected by k edge-disjoint paths.*
Theorem (Menger’s Theorem)

\[ G \text{ is } k\text{-connected, iff any two node are connected by } k \text{ intern-node-disjoint paths.} \]

Theorem (Menger's Theorem)

\[ G \text{ is } k\text{-edge-connected, iff any two node are connected by } k \text{ edge-disjoint paths.} \]
Computing the Connectivity

Theorem

The 1-connectivity of a graph may be computed by DFS/BFS.

Theorem

The 1-edge-connectivity of a graph may be computed by DFS/BFS.

Theorem

The 2-connectivity of a graph may be computed by DFS/BFS.

Theorem

The k-connectivity of a graph may be computed by flow algorithms.

Theorem

The k-edge-connectivity of a graph may be computed by flow algorithms.
The 1-connectivity of a graph may be computed by DFS/BFS.

The 1-edge-connectivity of a graph may be computed by DFS/BFS.

The 2-connectivity of a graph may be computed by DFS/BFS.

The k-connectivity of a graph may be computed by flow algorithms.

The k-edge-connectivity of a graph may be computed by flow algorithms.
Computing the Connectivity

Theorem

The 1-connectivity of a graph may be computed by DFS/BFS.

Theorem

The 1-edge-connectivity of a graph may be computed by DFS/BFS.

Theorem

The 2-connectivity of a graph may be computed by DFS/BFS.

Theorem

The k-connectivity of a graph may be computed by flow algorithms.

Theorem

The k-edge-connectivity of a graph may be computed by flow algorithms.
### Computing the Connectivity

<table>
<thead>
<tr>
<th>Theorem</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>The 1-connectivity of a graph may be computed by DFS/BFS.</strong></td>
</tr>
<tr>
<td><strong>The 1-edge-connectivity of a graph may be computed by DFS/BFS.</strong></td>
</tr>
<tr>
<td><strong>The 2-connectivity of a graph may be computed by DFS/BFS.</strong></td>
</tr>
<tr>
<td><strong>The k-connectivity of a graph may be computed by flow algorithms.</strong></td>
</tr>
<tr>
<td><strong>The k-edge-connectivity of a graph may be computed by flow algorithms.</strong></td>
</tr>
</tbody>
</table>
Computing the Connectivity

**Theorem**

The 1-connectivity of a graph may be computed by DFS/BFS.

**Theorem**

The 1-edge-connectivity of a graph may be computed by DFS/BFS.

**Theorem**

The 2-connectivity of a graph may be computed by DFS/BFS.

**Theorem**

The k-connectivity of a graph may be computed by flow algorithms.

**Theorem**

The k-edge-connectivity of a graph may be computed by flow algorithms.
Definition: Graph

Let $V(G) = \{v_1, ..., v_n\}$ be a non-empty set of nodes and

$E(G)$ a set or multiset of pairs from $V \times V$ (set of edges).

The sets $V(G)$ and $E(G)$ define the graph $G = (V(G), E(G))$.

If $G$ is uniquely determined, then we just write: $V$ and $E$.

Or in other words $G = (V, E)$.

We always use as default writing: $n = |V|$ and $m = |E|$.
Definition: Graph

Definition (Directed Graph)

- Let $V(G) = \{v_1, ..., v_n\}$ be a non-empty set of nodes and
- $E(G)$ a set or multiset of pairs from $V \times V$ (set of edges).
- The sets $V(G)$ and $E(G)$ define the graph $G = (V(G), E(G))$.
- If $G$ is uniquely determined, then we just write: $V$ and $E$.
- Or in other words $G = (V, E)$.
- We always use as default writing: $n = |V|$ and $m = |E|$.
Definition: Graph

- Let $V(G) = \{v_1, ..., v_n\}$ be a non-empty set of nodes and
- $E(G)$ a set or multiset of pairs from $V \times V$ (set of edges).
- The sets $V(G)$ and $E(G)$ define the graph $G = (V(G), E(G))$.
- If $G$ is uniquely determined, then we just write: $V$ and $E$.
- Or in other words $G = (V, E)$.
- We always use as default writing: $n = |V|$ and $m = |E|$.
Definition: Graph

**Definition (Directed Graph)**

- Let $V(G) = \{v_1, \ldots, v_n\}$ be a non-empty set of nodes and
- $E(G)$ a set or multiset of pairs from $V \times V$ (set of edges).
- The sets $V(G)$ and $E(G)$ define the graph $G = (V(G), E(G))$.
- If $G$ is uniquely determined, then we just write: $V$ and $E$.
- Or in other words $G = (V, E)$.
- We always use as default writing: $n = |V|$ and $m = |E|$.
Definition: Graph

Definition (Directed Graph)

Let $V(G) = \{v_1, ..., v_n\}$ be a non-empty set of nodes and

$E(G)$ a set or multiset of pairs from $V \times V$ (set of edges).

The sets $V(G)$ and $E(G)$ define the graph $G = (V(G), E(G))$.

If $G$ is uniquely determined, then we just write: $V$ and $E$.

Or in other words $G = (V, E)$.

We always use as default writing: $n = |V|$ and $m = |E|$. 
Definition: Graph

Let $V(G) = \{v_1, ..., v_n\}$ be a non-empty set of nodes and $E(G)$ a set or multiset of pairs from $V \times V$ (set of edges).

The sets $V(G)$ and $E(G)$ define the graph $G = (V(G), E(G))$.

If $G$ is uniquely determined, then we just write: $V$ and $E$.

Or in other words $G = (V, E)$.

We always use as default writing: $n = |V|$ and $m = |E|$.
Definition: Graph

Let $V(G) = \{v_1, \ldots, v_n\}$ be a non-empty set of nodes and $E(G)$ a set or multiset of pairs from $V \times V$ (set of edges).

The sets $V(G)$ and $E(G)$ define the graph $G = (V(G), E(G))$.

If $G$ is uniquely determined, then we just write: $V$ and $E$.

Or in other words $G = (V, E)$.

We always use as default writing: $n = |V|$ and $m = |E|$.
Strong Connectivity

**Definition**

A directed graph $G = (V, E)$ is called strongly connected, iff for any two different nodes $a, b$ exists a path from $a$ to $b$.

**Theorem**

The strong connectivity of a graph may be computed by DFS/BFS.
Definition

A directed graph $G = (V, E)$ is called strongly connected, iff for any two different nodes $a, b$ exists a path from $a$ to $b$.

Theorem

The strong connectivity of a graph may be computed by DFS/BFS.
The Flow Problem

**Definition (Flow)**

- Let $G = (V, E)$ a directed graph with cost-function $c : E \rightarrow \mathbb{N}$. Let $s, t \in V$ be the source and drain.
  - A function $f : E \rightarrow \mathbb{N}$ is a flow-function, iff
    - $\forall e \in E : 0 \leq f(e) \leq c(e)$
    - $\forall v \in V \setminus \{s, t\} : \sum_{e = (v, w) \in E} f(e) = \sum_{e = (w, v) \in E} f(e)$
  - The value of the flow is: $\sum_{e = (s, w) \in E} f(e) - \sum_{e = (w, s) \in E} f(e)$

**Definition (Maximal Flow Problem)**

Given: Graph $G = (V, E)$, $s, t \in V$ and $c : E \rightarrow \mathbb{N}$
Compute: Maximal flow-function $f$.

**Theorem (Maximal Flow Problem)**

*The problem to compute the maximal flow is in $\mathcal{P}$.*
The Flow Problem

Definition (Flow)

- Let $G = (V, E)$ a directed graph with cost-function $c : E \rightarrow \mathbb{N}$. Let $s, t \in V$ be the source and drain.

- A function $f : E \rightarrow \mathbb{N}$ is a flow-function, iff
  
  - $\forall e \in E : 0 \leq f(e) \leq c(e)$
  - $\forall v \in V \setminus \{s, t\} : \sum_{e=(v,w) \in E} f(e) = \sum_{e=(w,v) \in E} f(e)$

- The value of the flow is: $\sum_{e=(s,w) \in E} f(e) - \sum_{e=(w,s) \in E} f(e)$

Definition (Maximal Flow Problem)

Given: Graph $G = (V, E)$, $s, t \in V$ and $c : E \rightarrow \mathbb{N}$

Compute: Maximal flow-function $f$.

Theorem (Maximal Flow Problem)

The problem to compute the maximal flow is in $\mathcal{P}$. 


The Flow Problem

**Definition (Flow)**

- Let $G = (V, E)$ a directed graph with cost-function $c : E \mapsto \mathbb{N}$. Let $s, t \in V$ be the source and drain.
- A function $f : E \mapsto \mathbb{N}$ is a flow-function, iff
  - $\forall e \in E : 0 \leq f(e) \leq c(e)$
  - $\forall v \in V \setminus \{s, t\} : \sum_{e=(v,w)\in E} f(e) = \sum_{e=(w,v)\in E} f(e)$
- The value of the flow is: $\sum_{e=(s,w)\in E} f(e) - \sum_{e=(w,s)\in E} f(e)$

**Definition (Maximal Flow Problem)**

Given: Graph $G = (V, E)$, $s, t \in V$ and $c : E \mapsto \mathbb{N}$
Compute: Maximal flow-function $f$.

**Theorem (Maximal Flow Problem)**

*The problem to compute the maximal flow is in $\mathcal{P}$.*
The Flow Problem

Definition (Flow)

- Let $G = (V, E)$ a directed graph with cost-function $c : E \mapsto \mathbb{N}$. Let $s, t \in V$ be the source and drain.

- A function $f : E \mapsto \mathbb{N}$ is a flow-function, iff
  - $\forall e \in E : 0 \leq f(e) \leq c(e)$
  - $\forall v \in V \setminus \{s, t\} : \sum_{e=(v,w)\in E} f(e) = \sum_{e=(w,v)\in E} f(e)$

- The value of the flow is: $\sum_{e=(s,w)\in E} f(e) - \sum_{e=(w,s)\in E} f(e)$

Definition (Maximal Flow Problem)

Given: Graph $G = (V, E)$, $s, t \in V$ and $c : E \mapsto \mathbb{N}$

Compute: Maximal flow-function $f$.

Theorem (Maximal Flow Problem)

The problem to compute the maximal flow is in $\mathcal{P}$. 
The Flow Problem

Definition (Flow)

- Let $G = (V, E)$ a directed graph with cost-function $c : E \mapsto \mathbb{N}$. Let $s, t \in V$ be the source and drain.

- A function $f : E \mapsto \mathbb{N}$ is a flow-function, iff
  - $\forall e \in E : 0 \leq f(e) \leq c(e)$
  - $\forall v \in V \setminus \{s, t\} : \sum_{e=(v,w) \in E} f(e) = \sum_{e=(w,v) \in E} f(e)$

- The value of the flow is: $\sum_{e=(s,w) \in E} f(e) - \sum_{e=(w,s) \in E} f(e)$

Definition (Maximal Flow Problem)

Given: Graph $G = (V, E)$, $s, t \in V$ and $c : E \mapsto \mathbb{N}$
Compute: Maximal flow-function $f$.

Theorem (Maximal Flow Problem)

The problem to compute the maximal flow is in $\mathcal{P}$. 
The Flow Problem

Definition (Flow)

- Let $G = (V, E)$ a directed graph with cost-function $c : E \mapsto \mathbb{N}$. Let $s, t \in V$ be the source and drain.

- A function $f : E \mapsto \mathbb{N}$ is a flow-function, iff
  - $\forall e \in E : 0 \leq f(e) \leq c(e)$
  - $\forall v \in V \setminus \{s, t\} : \sum_{e=(v,w) \in E} f(e) = \sum_{e=(w,v) \in E} f(e)$

- The value of the flow is: $\sum_{e=(s,w) \in E} f(e) - \sum_{e=(w,s) \in E} f(e)$

Definition (Maximal Flow Problem)

Given: Graph $G = (V, E)$, $s, t \in V$ and $c : E \mapsto \mathbb{N}$
Compute: Maximal flow-function $f$.

Theorem (Maximal Flow Problem)

The problem to compute the maximal flow is in $\mathcal{P}$. 
The Flow Problem

Definition (Flow)

- Let $G = (V, E)$ a directed graph with cost-function $c : E \mapsto \mathbb{N}$. Let $s, t \in V$ be the source and drain.
- A function $f : E \mapsto \mathbb{N}$ is a flow-function, iff
  - $\forall e \in E : 0 \leq f(e) \leq c(e)$
  - $\forall v \in V \setminus \{s, t\} : \sum_{e=(v,w) \in E} f(e) = \sum_{e=(w,v) \in E} f(e)$
- The value of the flow is: $\sum_{e=(s,w) \in E} f(e) - \sum_{e=(w,s) \in E} f(e)$

Definition (Maximal Flow Problem)

Given: Graph $G = (V, E), s, t \in V$ and $c : E \mapsto \mathbb{N}$
Compute: Maximal flow-function $f$.

Theorem (Maximal Flow Problem)

*The problem to compute the maximal flow is in $\mathcal{P}$.*
**The Flow Problem**

**Definition (Flow)**
- Let $G = (V, E)$ a directed graph with cost-function $c : E \rightarrow \mathbb{N}$. Let $s, t \in V$ be the source and drain.
- A function $f : E \rightarrow \mathbb{N}$ is a flow-function, iff
  - $\forall e \in E : 0 \leq f(e) \leq c(e)$
  - $\forall v \in V \setminus \{s, t\} : \sum_{e=(v,w)\in E} f(e) = \sum_{e=(w,v)\in E} f(e)$
- The value of the flow is: $\sum_{e=(s,w)\in E} f(e) - \sum_{e=(w,s)\in E} f(e)$

**Definition (Maximal Flow Problem)**
Given: Graph $G = (V, E)$, $s, t \in V$ and $c : E \rightarrow \mathbb{N}$
Compute: Maximal flow-function $f$.

**Theorem (Maximal Flow Problem)**
*The problem to compute the maximal flow is in $\mathcal{P}$.***
Minimal Cut

**Definition (Cut)**
- Let $G = (V, E)$ be a directed graph with cost-function $c : E \rightarrow \mathbb{N}$
- Let $s, t \in V$ source and drain.
- $A, B \subset V$ are called a cut, iff
  - $s \in A$ and $t \in B$
  - $A \cap B = \emptyset$ and $A \cup B = V$
- The capacity of the cut $A, B$ is: $\sum_{e=(v,w) \in E, v \in A, w \in B} c(e)$

**Theorem (Min-Cut-Max-Flow)**

The capacity of the minimal cut is the same as the maximal flow.
**Definition (Cut)**

- Let $G = (V, E)$ be a directed graph with cost-function $c : E \rightarrow \mathbb{N}$
- Let $s, t \in V$ source and drain.
- $A, B \subset V$ are called a cut, iff
  - $s \in A$ and $t \in B$
  - $A \cap B = \emptyset$ and $A \cup B = V$
- The capacity of the cut $A, B$ is: $\sum_{e=(v,w) \in E, v \in A, w \in B} c(e)$

**Theorem (Min-Cut-Max-Flow)**

*The capacity of the minimal cut is the same as the maximal flow.*
**Definition (Cut)**

- Let $G = (V, E)$ be a directed graph with cost-function $c : E \mapsto \mathbb{N}$
- Let $s, t \in V$ source and drain.
- $A, B \subseteq V$ are called a cut, iff
  - $s \in A$ and $t \in B$
  - $A \cap B = \emptyset$ and $A \cup B = V$
- The capacity of the cut $A, B$ is: $\sum_{e=(v,w) \in E, v \in A, w \in B} c(e)$

**Theorem (Min-Cut-Max-Flow)**

*The capacity of the minimal cut is the same as the maximal flow.*
Defintion (Cut)

- Let $G = (V, E)$ be a directed graph with cost-function $c : E \to \mathbb{N}$
- Let $s, t \in V$ source and drain.
- $A, B \subset V$ are called a cut, iff
  - $s \in A$ and $t \in B$
  - $A \cap B = \emptyset$ and $A \cup B = V$
- The capacity of the cut $A, B$ is: $\sum_{e = (v, w) \in E, v \in A, w \in B} c(e)$

Theorem (Min-Cut-Max-Flow)

*The capacity of the minimal cut is the same as the maximal flow.*
Definition (Cut)

- Let $G = (V, E)$ be a directed graph with cost-function $c : E \rightarrow \mathbb{N}$
- Let $s, t \in V$ source and drain.
- $A, B \subseteq V$ are called a cut, iff
  - $s \in A$ and $t \in B$
  - $A \cap B = \emptyset$ and $A \cup B = V$
- The capacity of the cut $A, B$ is: $\sum_{e = (v, w) \in E, v \in A, w \in B} c(e)$

Theorem (Min-Cut-Max-Flow)

The capacity of the minimal cut is the same as the maximal flow.
**Minimal Cut**

**Definition (Cut)**

- Let $G = (V, E)$ be a directed graph with cost-function $c : E \mapsto \mathbb{N}$
- Let $s, t \in V$ source and drain.
- $A, B \subset V$ are called a cut, iff
  - $s \in A$ and $t \in B$
  - $A \cap B = \emptyset$ and $A \cup B = V$
- **The capacity of the cut $A, B$ is:** $\sum_{e=(v,w)\in E, v\in A, w\in B} c(e)$

**Theorem (Min-Cut-Max-Flow)**

*The capacity of the minimal cut is the same as the maximal flow.*
Definition (Cut)

Let $G = (V, E)$ be a directed graph with cost-function $c : E \to \mathbb{N}$

Let $s, t \in V$ source and drain.

$A, B \subseteq V$ are called a cut, iff

$s \in A$ and $t \in B$

$A \cap B = \emptyset$ and $A \cup B = V$

The capacity of the cut $A, B$ is: $\sum_{e=(v,w) \in E, v \in A, w \in B} c(e)$

Theorem (Min-Cut-Max-Flow)

The capacity of the minimal cut is the same as the maximal flow.
Minimal Cut

**Definition (Cut)**

- Let $G = (V, E)$ be a directed graph with cost-function $c : E \rightarrow \mathbb{N}$
- Let $s, t \in V$ source and drain.
- $A, B \subset V$ are called a cut, iff
  - $s \in A$ and $t \in B$
  - $A \cap B = \emptyset$ and $A \cup B = V$
- The capacity of the cut $A, B$ is: $\sum_{e=(v,w)\in E, v\in A, w\in B} c(e)$

**Theorem (Min-Cut-Max-Flow)**

*The capacity of the minimal cut is the same as the maximal flow.*
Maximal Matching Problem

Definition

Let $G = (V, E)$ be a graph. The edges $e, e' \in E$ are called independent, iff they share no common node.

Definition (Matching)

Let $G = (V, E)$ be a graph. $M \subseteq E$ is called a matching, iff $\forall e, f \in M, e \neq f : e \cap f = \emptyset$. $M$ is a set of independent edges.

Definition

Let $G = (V_1, V_2, E)$ be a bipartite graph, and there exists a set $M$ of $|V_1|$ independent edges. We call $M$ complete matching from $V_1$ to $V_2$. 
Maximal Matching Problem

Definition

Let \( G = (V, E) \) be a graph. The edges \( e, e' \in E \) are called independent, iff they share no common node.

Definition (Matching)

Let \( G = (V, E) \) be a graph.
\( M \subseteq E \) is called a matching, iff \( \forall e, f \in M, e \neq f : e \cap f = \emptyset \).
\( M \) is a set of independent edges.

Definition

Let \( G = (V_1, V_2, E) \) be a bipartite graph, and there exists a set \( M \) of \( |V_1| \) independent edges. We call \( M \) a complete matching from \( V_1 \) to \( V_2 \).
Maximal Matching Problem

**Definition**
Let $G = (V, E)$ be a graph. The edges $e, e' \in E$ are called **independent**, iff they share no common node.

**Definition (Matching)**
Let $G = (V, E)$ be a graph. $M \subseteq E$ is called a matching, iff $\forall e, f \in M, e \neq f : e \cap f = \emptyset$. $M$ is a set of independent edges.

**Definition**
Let $G = (V_1, V_2, E)$ be a bipartite graph, and there exists a set $M$ of $|V_1|$ independent edges. We call $M$ **complete matching** from $V_1$ to $V_2$. 
Theorem of Hall

**Definition**

Let $G = (V_1, V_2, E)$ be a bipartite graph, and $A \subseteq V_1$. We denote:

$$\Gamma(A) = \{v \in V_2 \mid (v, w) \in E, w \in A\}.$$

**Theorem (Hall)**

Let $G = (V_1, V_2, E)$ be a bipartite graph. There exits a complete matching from $V_1$ to $V_2$, iff for each $A \subseteq V_1$ we have

$$|\Gamma(A)| \geq |A|.$$ 

**Corollary**

Every regular bipartite Graph $G = (V_1, V_2, E)$ with $|V_1| = |V_2|$ contains a complete matching.
Theorem of Hall

Definition

Let $G = (V_1, V_2, E)$ be a bipartite graph, and $A \subseteq V_1$. We denote:

$$\Gamma(A) = \{ v \in V_2 \mid (v, w) \in E, w \in A \}.$$  

Theorem (Hall)

Let $G = (V_1, V_2, E)$ be a bipartite graph. There exists a complete matching from $V_1$ to $V_2$, iff for each $A \subseteq V_1$ we have

$$|\Gamma(A)| \geq |A|.$$  

Corollary

Every regular bipartite graph $G = (V_1, V_2, E)$ with $|V_1| = |V_2|$ contains a complete matching.
Theorem of Hall

**Definition**

Let $G = (V_1, V_2, E)$ be a bipartite graph, and $A \subseteq V_1$. We denote:

$$\Gamma(A) = \{ v \in V_2 \mid (v, w) \in E, w \in A \}.$$ 

**Theorem (Hall)**

Let $G = (V_1, V_2, E)$ be a bipartite graph. There exits a complete matching from $V_1$ to $V_2$, iff for each $A \subseteq V_1$ we have

$$|\Gamma(A)| \geq |A|.$$ 

**Corollary**

Every regular bipartite Graph $G = (V_1, V_2, E)$ with $|V_1| = |V_2|$ contains a complete matching.
Proof (Hall)

Theorem (Hall)

Let $G = (V_1, V_2, E)$ be a bipartite graph. There exists a complete matching from $V_1$ to $V_2$, iff for each $A \subseteq V_1$ we have

$$|\Gamma(A)| \geq |A|.$$ 

$\implies$ simple:

- Let $M$ be a matching with $|M| = |V_1|$ and let $A \subset V_1$ arbitrary.
- $|\Gamma(A)| = |\{v \in V_2 \mid (v, w) \in E, w \in A\}|.$
- $|\Gamma(A)| \geq |\{v \in V_2 \mid (v, w) \in M, w \in A\}|.$
- $|\Gamma(A)| \geq |A|.$
Proof (Hall)

Theorem (Hall)

Let $G = (V_1, V_2, E)$ be a bipartite graph. There exists a complete matching from $V_1$ to $V_2$, iff for each $A \subseteq V_1$ we have

$$|\Gamma(A)| \geq |A|.$$
Proof (Hall)

**Theorem (Hall)**

Let $G = (V_1, V_2, E)$ be a bipartite graph. There exits a complete matching from $V_1$ to $V_2$, iff for each $A \subseteq V_1$ we have

$$|\Gamma(A)| \geq |A|.$$ 

---

**Simple:**

- Let $M$ be a matching with $|M| = |V_1|$ and let $A \subseteq V_1$ arbitrary.
- $|\Gamma(A)| = |\{v \in V_2 \mid (v, w) \in E, w \in A\}|$.
- $|\Gamma(A)| \geq |\{v \in V_2 \mid (v, w) \in M, w \in A\}|$.
- $|\Gamma(A)| \geq |A|$.
Proof (Hall)

**Theorem (Hall)**

Let $G = (V_1, V_2, E)$ be a bipartite graph. There exits a complete matching from $V_1$ to $V_2$, iff for each $A \subseteq V_1$ we have

$$|\Gamma(A)| \geq |A|.$$ 

 ⇒ simple:

- Let $M$ be a matching with $|M| = |V_1|$ and let $A \subseteq V_1$ arbitrary.
- $|\Gamma(A)| = |\{v \in V_2 \mid (v, w) \in E, w \in A\}|.$
- $|\Gamma(A)| \geq |\{v \in V_2 \mid (v, w) \in M, w \in A\}|.$
- $|\Gamma(A)| \geq |A|.$
Proof (Hall)

Theorem (Hall)

Let $G = (V_1, V_2, E)$ be a bipartite graph. There exists a complete matching from $V_1$ to $V_2$, iff for each $A \subseteq V_1$ we have

$$|\Gamma(A)| \geq |A|.$$  

\[ \implies \text{ simple:} \]

- Let $M$ be a matching with $|M| = |V_1|$ and let $A \subset V_1$ arbitrary.
- $|\Gamma(A)| = |\{v \in V_2 \mid (v, w) \in E, w \in A\}|$.
- $|\Gamma(A)| \geq |\{v \in V_2 \mid (v, w) \in M, w \in A\}|$.
- $|\Gamma(A)| \geq |A|$.
Proof (Hall)

Theorem (Hall)

Let \( G = (V_1, V_2, E) \) be a bipartite graph. There exits a complete matching from \( V_1 \) to \( V_2 \), iff for each \( A \subseteq V_1 \) we have

\[
|\Gamma(A)| \geq |A|.
\]

\( \implies \) simple:

- Let \( M \) be a matching with \( |M| = |V_1| \) and let \( A \subseteq V_1 \) arbitrary.
- \( |\Gamma(A)| = |\{ v \in V_2 \mid (v, w) \in E, w \in A \}|. \)
- \( |\Gamma(A)| \geq |\{ v \in V_2 \mid (v, w) \in M, w \in A \}|. \)
- \( |\Gamma(A)| \geq |A|. \)
Proof (Hall)

Theorem (Hall)

Let $G = (V_1, V_2, E)$ be a bipartite graph. There exits a complete matching from $V_1$ to $V_2$, iff for each $A \subseteq V_1$ we have

$$|\Gamma(A)| \geq |A|.$$ 

$\Leftarrow$ by contradiction:

- Let $M$ be the largest matching with $|M| < |V_1|$.
- Let $A_1 = \{v \in V_1 | \exists b \in V_2 : \{v, b\} \in M\}$.
- Let $A_2 = \{v \in V_2 | \exists b \in V_1 : \{v, b\} \in M\}$.
- Let $a \in V_1 \setminus A_1$.
- $\Gamma(a) \subseteq A_2$, because $M$ is the largest matching.
- Any alternating path starting from $a$ reaches only nodes in $A_1' \cup A_2'$ with $A_i' \subseteq A_i$ and $|A_i'| = |A_i|$
- Thus we have $\Gamma(A_1' \cup \{a\}) \subseteq A_2'$.
- $|A_1' \cup \{a\}| > |A_2'|$. 
**Theorem (Hall)**

Let $G = (V_1, V_2, E)$ be a bipartite graph. There exits a complete matching from $V_1$ to $V_2$, iff for each $A \subseteq V_1$ we have

$$|\Gamma(A)| \geq |A|.$$  

<--- by contradiction:

- Let $M$ be the largest matching with $|M| < |V_1|$.
- Let $A_1 = \{v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in M\}$.
- Let $A_2 = \{v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in M\}$.
- Let $a \in V_1 \setminus A_1$.
- $\Gamma(a) \subset A_2$, because $M$ is the largest matching.
- Any alternating path starting from $a$ reaches only nodes in $A'_1 \cup A'_2$ with $A'_i \subset A_i$ and $|A'_1| = |A'_2|$.
- Thus we have $\Gamma(A'_1 \cup \{a\}) \subset A'_2$.
- $|A'_1 \cup \{a\}| > |A'_2|$. 
Proof (Hall)

Theorem (Hall)

Let $G = (V_1, V_2, E)$ be a bipartite graph. There exists a complete matching from $V_1$ to $V_2$, iff for each $A \subseteq V_1$ we have

$$|\Gamma(A)| \geq |A|.$$

\[\iff\]

by contradiction:

- Let $M$ be the largest matching with $|M| < |V_1|$.
- Let $A_1 = \{ v \in V_1 | \exists b \in V_2 : \{v, b\} \in M \}$.
- Let $A_2 = \{ v \in V_2 | \exists b \in V_1 : \{v, b\} \in M \}$.
- Let $a \in V_1 \setminus A_1$.
- $\Gamma(a) \subseteq A_2$, because $M$ is the largest matching.
- Any alternating path starting from $a$ reaches only nodes in $A'_1 \cup A'_2$ with $A'_i \subseteq A_i$ and $|A'_1| = |A'_2|$.
- Thus we have $\Gamma(A'_1 \cup \{a\}) \subseteq A'_2$.
- $|A'_1 \cup \{a\}| > |A'_2|$.
Theorem (Hall)

Let $G = (V_1, V_2, E)$ be a bipartite graph. There exists a complete matching from $V_1$ to $V_2$, iff for each $A \subseteq V_1$ we have

$$|\Gamma(A)| \geq |A|.$$ 

$\iff$ by contradiction:

- Let $M$ be the largest matching with $|M| < |V_1|$.
- Let $A_1 = \{ v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in M \}$.
- Let $A_2 = \{ v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in M \}$.
- Let $a \in V_1 \setminus A_1$.
- $\Gamma(a) \subset A_2$, because $M$ is the largest matching.
- Any alternating path starting from $a$ reaches only nodes in $A_1' \cup A_2'$ with $A_i' \subset A_i$ and $|A_i'| = |A_i'|$.
- Thus we have $\Gamma(A_1' \cup \{a\}) \subset A_2'$.
- $|A_1' \cup \{a\}| > |A_2'|$. 


Proof (Hall)

Theorem (Hall)

Let $G = (V_1, V_2, E)$ be a bipartite graph. There exists a complete matching from $V_1$ to $V_2$, iff for each $A \subseteq V_1$ we have

$$|\Gamma(A)| \geq |A|.$$ 

by contradiction:

- Let $M$ be the largest matching with $|M| < |V_1|$.
- Let $A_1 = \{ v \in V_1 \mid \exists b \in V_2 : \{ v, b \} \in M \}$.
- Let $A_2 = \{ v \in V_2 \mid \exists b \in V_1 : \{ v, b \} \in M \}$.
- Let $a \in V_1 \setminus A_1$.
- $\Gamma(a) \subseteq A_2$, because $M$ is the largest matching.
- Any alternating path starting from $a$ reaches only nodes in $A_1' \cup A_2'$ with $A_i' \subset A_i$ and $|A_i'| = |A_i'|$.
- Thus we have $\Gamma(A_1' \cup \{ a \}) \subseteq A_2'$.
- $|A_1' \cup \{ a \}| > |A_2'|$. 

Proof (Hall)

**Theorem (Hall)**

Let $G = (V_1, V_2, E)$ be a bipartite graph. There exits a complete matching from $V_1$ to $V_2$, iff for each $A \subseteq V_1$ we have

$$|\Gamma(A)| \geq |A|.$$ 

\[\Leftarrow\] by contradiction:

- Let $M$ be the largest matching with $|M| < |V_1|$.
- Let $A_1 = \{v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in M\}$.
- Let $A_2 = \{v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in M\}$.
- Let $a \in V_1 \setminus A_1$.
- $\Gamma(a) \subset A_2$, because $M$ is the largest matching.
- Any alternating path starting from $a$ reaches only nodes in $A'_1 \cup A'_2$ with $A'_i \subset A_i$ and $|A'_i| = |A'_2|$.
- Thus we have $\Gamma(A'_1 \cup \{a\}) \subset A'_2$.
- $|A'_1 \cup \{a\}| > |A'_2|$.
Proof (Hall)

**Theorem (Hall)**

Let $G = (V_1, V_2, E)$ be a bipartite graph. There exists a complete matching from $V_1$ to $V_2$, iff for each $A \subseteq V_1$ we have

$$|\Gamma(A)| \geq |A|.$$ 

\[\iff\]

by contradiction:

- Let $M$ be the largest matching with $|M| < |V_1|$.
- Let $A_1 = \{v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in M\}$.
- Let $A_2 = \{v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in M\}$.
- Let $a \in V_1 \setminus A_1$.
- $\Gamma(a) \subset A_2$, because $M$ is the largest matching.
- Any alternating path starting from $a$ reaches only nodes in $A_1' \cup A_2'$ with $A_i' \subset A_i$ and $|A_1'| = |A_2'|$.
- Thus we have $\Gamma(A_1' \cup \{a\}) \subset A_2'$.
- $|A_1' \cup \{a\}| > |A_2'|.$
Let $G = (V_1, V_2, E)$ be a bipartite graph. There exists a complete matching from $V_1$ to $V_2$, iff for each $A \subseteq V_1$ we have $|\Gamma(A)| \geq |A|$. 

\[ \iff \]

\begin{itemize}
  \item Let $M$ be the largest matching with $|M| < |V_1|$.
  \item Let $A_1 = \{v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in M\}$.
  \item Let $A_2 = \{v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in M\}$.
  \item Let $a \in V_1 \setminus A_1$.
  \item $\Gamma(a) \subset A_2$, because $M$ is the largest matching.
  \item Any alternating path starting from $a$ reaches only nodes in $A'_1 \cup A'_2$ with $A'_i \subset A_i$ and $|A'_i| = |A'_2|$.
  \item Thus we have $\Gamma(A'_1 \cup \{a\}) \subset A'_2$.
  \item $|A'_1 \cup \{a\}| > |A'_2|$.
\end{itemize}
Proof (Hall)

Theorem (Hall)

Let \( G = (V_1, V_2, E) \) be a bipartite graph. There exits a complete matching from \( V_1 \) to \( V_2 \), iff for each \( A \subseteq V_1 \) we have

\[ |\Gamma(A)| \geq |A|. \]

\[ \Leftarrow \] by contradiction:

- Let \( M \) be the largest matching with \( |M| < |V_1| \).
- Let \( A_1 = \{ v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in M \} \).
- Let \( A_2 = \{ v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in M \} \).
- Let \( a \in V_1 \setminus A_1 \).
- \( \Gamma(a) \subseteq A_2 \), because \( M \) is the largest matching.
- Any alternating path starting from \( a \) reaches only nodes in \( A_1' \cup A_2' \) with \( A_i' \subseteq A_i \) and \( |A_i'| = |A_i'| \).
- Thus we have \( \Gamma(A_1' \cup \{a\}) \subseteq A_2' \).
- \( |A_1' \cup \{a\}| > |A_2'| \).
Proof (Hall)

**Theorem (Hall)**

Let $G = (V_1, V_2, E)$ be a bipartite graph. There exists a complete matching from $V_1$ to $V_2$, iff for each $A \subseteq V_1$ we have

$$|\Gamma(A)| \geq |A|.$$ 

by contradiction:

- Let $M$ be the largest matching with $|M| < |V_1|$.
- Let $A_1 = \{v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in M\}$.
- Let $A_2 = \{v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in M\}$.
- Let $a \in V_1 \setminus A_1$.
- $\Gamma(a) \subseteq A_2$, because $M$ is the largest matching.
- Any alternating path starting from $a$ reaches only nodes in $A_1' \cup A_2'$ with $A_i' \subseteq A_i$ and $|A_i'| = |A_2'|$.
- Thus we have $\Gamma(A_1' \cup \{a\}) \subseteq A_2'$.
- $|A_1' \cup \{a\}| > |A_2'|$. 

Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph and $|\Gamma(A)| \geq |A| - d$ for every $A \subseteq V_1$. Then contains $G$ at least $|V_1| - d$ independent edges.

$\implies$ By contradiction:

1. Let $M$ be the largest matching with $m = |M| < |V_1| - d$.
2. Let $A_1 = \{v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in M\}$.
3. Let $A_2 = \{v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in M\}$.
4. Let $a_0, a_1, \cdots, a_d \in V_1 \setminus A_1$.
5. $N(a_i) \subseteq A_2$, because $M$ is the largest matching.
6. Any alternating path starting from $a_i$ reaches only nodes in $A_1 \cup A_2$.
7. Thus we get $\Gamma(A_1 \cup \{a_i\}) \subseteq A_2$.
8. $m + d + 1 = |A_1 \cup \{a_i \mid 0 \leq i \leq d\}| \geq |A_2| = m$. 
Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph and $|\Gamma(A)| \geq |A| - d$ for every $A \subseteq V_1$. Then contains $G$ at least $|V_1| - d$ independent edges.

$\implies$ By contradiction:

- Let $M$ be the largest matching with $m = |M| < |V_1| - d$.
- Let $A_1 = \{v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in M\}$.
- Let $A_2 = \{v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in M\}$.
- Let $a_0, a_1, \cdots, a_d \in V_1 \setminus A_1$.
- $N(a_i) \subset A_2$, because $M$ is the largest matching.
- Any alternating path starting from $a_i$ reaches only nodes in $A_1 \cup A_2$.
- Thus we get $\Gamma(A_1 \cup \{a_i\}) \subset A_2$.
- $m + d + 1 = |A_1 \cup \{a_i \mid 0 \leq i \leq d\}| \geq |A_2| = m$. 
Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph and $|\Gamma(A)| \geq |A| - d$ for every $A \subseteq V_1$. Then contains $G$ at least $|V_1| - d$ independent edges.

By contradiction:

- Let $M$ be the largest matching with $m = |M| < |V_1| - d$.
- Let $A_1 = \{v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in M\}$.
- Let $A_2 = \{v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in M\}$.
- Let $a_0, a_1, \cdots, a_d \in V_1 \setminus A_1$.
- $N(a_i) \subseteq A_2$, because $M$ is the largest matching.
- Any alternating path starting from $a_i$ reaches only nodes in $A_1 \cup A_2$.
- Thus we get $\Gamma(A_1 \cup \{a_i\}) \subseteq A_2$.
- $m + d + 1 = |A_1 \cup \{a_i \mid 0 \leq i \leq d\}| \geq |A_2| = m$. 
Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph and $|\Gamma(A)| \geq |A| - d$ for every $A \subseteq V_1$. Then contains $G$ at least $|V_1| - d$ independent edges.

$\implies$ By contradiction:

- Let $M$ be the largest matching with $m = |M| < |V_1| - d$.
- Let $A_1 = \{v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in M\}$.
- Let $A_2 = \{v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in M\}$.
- Let $a_0, a_1, \cdots, a_d \in V_1 \setminus A_1$.
- $N(a_i) \subseteq A_2$, because $M$ is the largest matching.
- Any alternating path starting from $a_i$ reaches only nodes in $A_1 \cup A_2$.
- Thus we get $\Gamma(A_1 \cup \{a_i\}) \subseteq A_2$.
- $m + d + 1 = |A_1 \cup \{a_i \mid 0 \leq i \leq d\}| \geq |A_2| = m$. 
Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph and $|\Gamma(A)| \geq |A| - d$ for every $A \subseteq V_1$. Then contains $G$ at least $|V_1| - d$ independent edges.

By contradiction:

- Let $M$ be the largest matching with $m = |M| < |V_1| - d$.
- Let $A_1 = \{v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in M\}$.
- Let $A_2 = \{v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in M\}$.
- Let $a_0, a_1, \ldots, a_d \in V_1 \setminus A_1$.
- $N(a_i) \subseteq A_2$, because $M$ is the largest matching.
- Any alternating path starting from $a_i$ reaches only nodes in $A_1 \cup A_2$.
- Thus we get $\Gamma(A_1 \cup \{a_i\}) \subseteq A_2$.
- $m + d + 1 = |A_1 \cup \{a_i \mid 0 \leq i \leq d\}| \geq |A_2| = m$. 
Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph and $|\Gamma(A)| \geq |A| - d$ for every $A \subseteq V_1$. Then contains $G$ at least $|V_1| - d$ independent edges.

\[ \Rightarrow \quad \text{By contradiction:} \]

- Let $M$ be the largest matching with $m = |M| < |V_1| - d$.
- Let $A_1 = \{ v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in M \}$.
- Let $A_2 = \{ v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in M \}$.
- Let $a_0, a_1, \ldots, a_d \in V_1 \setminus A_1$.
- $N(a_i) \subseteq A_2$, because $M$ is the largest matching.
- Any alternating path starting from $a_i$ reaches only nodes in $A_1 \cup A_2$.
- Thus we get $\Gamma(A_1 \cup \{a_i\}) \subseteq A_2$.
- $m + d + 1 = |A_1 \cup \{a_i \mid 0 \leq i \leq d\}| \geq |A_2| = m$. 
Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph and $|\Gamma(A)| \geq |A| - d$ for every $A \subseteq V_1$. Then contains $G$ at least $|V_1| - d$ independent edges.

By contradiction:

- Let $M$ be the largest matching with $m = |M| < |V_1| - d$.
- Let $A_1 = \{v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in M\}$.
- Let $A_2 = \{v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in M\}$.
- Let $a_0, a_1, \cdots, a_d \in V_1 \setminus A_1$.
- $N(a_i) \subseteq A_2$, because $M$ is the largest matching.
- Any alternating path starting from $a_i$ reaches only nodes in $A_1 \cup A_2$.
- Thus we get $\Gamma(A_1 \cup \{a_i\}) \subseteq A_2$.
- $m + d + 1 = |A_1 \cup \{a_i \mid 0 \leq i \leq d\}| \geq |A_2| = m$. 
Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph and $|\Gamma(A)| \geq |A| - d$ for every $A \subseteq V_1$. Then contains $G$ at least $|V_1| - d$ independent edges.

⇒ By contradiction:

- Let $M$ be the largest matching with $m = |M| < |V_1| - d$.
- Let $A_1 = \{v \in V_1 | \exists b \in V_2 : \{v, b\} \in M\}$.
- Let $A_2 = \{v \in V_2 | \exists b \in V_1 : \{v, b\} \in M\}$.
- Let $a_0, a_1, \cdots, a_d \in V_1 \setminus A_1$.
- $N(a_i) \subseteq A_2$, because $M$ is the largest matching.
- Any alternating path starting from $a_i$ reaches only nodes in $A_1 \cup A_2$.
- Thus we get $\Gamma(A_1 \cup \{a_i\}) \subseteq A_2$.
- $m + d + 1 = |A_1 \cup \{a_i \mid 0 \leq i \leq d\}| \geq |A_2| = m$. 
Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph and $|\Gamma(A)| \geq |A| - d$ for every $A \subseteq V_1$. Then contains $G$ at least $|V_1| - d$ independent edges.

By contradiction:

1. Let $M$ be the largest matching with $m = |M| < |V_1| - d$.
2. Let $A_1 = \{v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in M\}$.
3. Let $A_2 = \{v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in M\}$.
4. Let $a_0, a_1, \ldots, a_d \in V_1 \setminus A_1$.
5. $N(a_i) \subseteq A_2$, because $M$ is the largest matching.
6. Any alternating path starting from $a_i$ reaches only nodes in $A_1 \cup A_2$.
7. Thus we get $\Gamma(A_1 \cup \{a_i\}) \subseteq A_2$.
8. $m + d + 1 = |A_1 \cup \{a_i \mid 0 \leq i \leq d\}| \geq |A_2| = m$. 
Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph and $|\Gamma(A)| \geq |A| - d$ for every $A \subseteq V_1$. Then contains $G$ at least $|V_1| - d$ independent edges.

$\implies$ By contradiction:

- Let $M$ be the largest matching with $m = |M| < |V_1| - d$.
- Let $A_1 = \{v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in M\}$.
- Let $A_2 = \{v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in M\}$.
- Let $a_0, a_1, \ldots, a_d \in V_1 \setminus A_1$.
- $N(a_i) \subseteq A_2$, because $M$ is the largest matching.
- Any alternating path starting from $a_i$ reaches only nodes in $A_1 \cup A_2$.
- Thus we get $\Gamma(A_1 \cup \{a_i\}) \subseteq A_2$.
- $m + d + 1 = |A_1 \cup \{a_i \mid 0 \leq i \leq d\}| \geq |A_2| = m$. 

**Corollary**

Let $G = (V_1, V_2, E)$ be a bipartite graph and $|\Gamma(A)| \geq |A| - d$ for every $A \subseteq V_1$. Then contains $G$ at least $|V_1| - d$ independent edges.

$\implies$ By contradiction:

- Let $M$ be the largest matching with $m = |M| < |V_1| - d$.
- Let $A_1 = \{v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in M\}$.
- Let $A_2 = \{v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in M\}$.
- Let $a_0, a_1, \ldots, a_d \in V_1 \setminus A_1$.
- $N(a_i) \subseteq A_2$, because $M$ is the largest matching.
- Any alternating path starting from $a_i$ reaches only nodes in $A_1 \cup A_2$.
- Thus we get $\Gamma(A_1 \cup \{a_i\}) \subset A_2$.
- $m + d + 1 = |A_1 \cup \{a_i \mid 0 \leq i \leq d\}| \geq |A_2| = m$. 
Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph with $V_1 = (x_1, ..., x_m)$ and $V_2 = (y_1, ..., y_n)$. Then contains $G$ a spanning graph $H$ with $\deg_H(x_i) = d_i$ and $0 \leq \deg_H(y_i) \leq 1$, iff for each $A \subseteq V_1$ we get

$$|\Gamma(A)| \geq \sum_{x_i \in A} d_i.$$

\[ \implies \text{simple:} \]

- Let $S$ be a spanning graph with $|S| = |V_1|$.
- Let $A \subseteq V_1$ arbitrary.
- $|\Gamma(A)| = |\{v \in V_2 \mid (v, w) \in E, w \in A\}|$.
- $|\Gamma(A)| \geq |\{v \in V_2 \mid (v, w) \in S, w \in A\}|$.
- $|\Gamma(A)| \geq \sum_{x_i \in A} d_i$. 
Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph with $V_1 = (x_1, ..., x_m)$ and $V_2 = (y_1, ..., y_n)$. Then contains $G$ a spanning graph $H$ with $\deg_H(x_i) = d_i$ and $0 \leq \deg_H(y_i) \leq 1$, iff for each $A \subseteq V_1$ we get

$$|\Gamma(A)| \geq \sum_{x_i \in A} d_i.$$


\[\implies\text{simple:}\]

- Let $S$ be a spanning graph with $|S| = |V_1|$.
- Let $A \subset V_1$ arbitrary.
- $|\Gamma(A)| = |\{v \in V_2 \mid (v, w) \in E, w \in A\}|$.
- $|\Gamma(A)| \geq |\{v \in V_2 \mid (v, w) \in S, w \in A\}|$.
- $|\Gamma(A)| \geq \sum_{x_i \in A} d_i$. 
Applications II

**Corollary**

Let $G = (V_1, V_2, E)$ be a bipartite graph with $V_1 = (x_1, \ldots, x_m)$ and $V_2 = (y_1, \ldots, y_n)$. Then contains $G$ a spanning graph $H$ with $\deg_H(x_i) = d_i$ and $0 \leq \deg_H(y_i) \leq 1$, iff for each $A \subseteq V_1$ we get

$$|\Gamma(A)| \geq \sum_{x_i \in A} d_i.$$ 

\[\Rightarrow\] simple:

- Let $S$ be a spanning graph with $|S| = |V_1|$.
- Let $A \subseteq V_1$ arbitrary.
- $|\Gamma(A)| = |\{v \in V_2 \mid (v, w) \in E, w \in A\}|$.
- $|\Gamma(A)| \geq |\{v \in V_2 \mid (v, w) \in S, w \in A\}|$.
- $|\Gamma(A)| \geq \sum_{x_i \in A} d_i$. 
Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph with $V_1 = (x_1, \ldots, x_m)$ and $V_2 = (y_1, \ldots, y_n)$. Then contains $G$ a spanning graph $H$ with $\deg_H(x_i) = d_i$ and $0 \leq \deg_H(y_i) \leq 1$, iff for each $A \subseteq V_1$ we get

$$|\Gamma(A)| \geq \sum_{x_i \in A} d_i.$$

.simple:

- Let $S$ be a spanning graph with $|S| = |V_1|$.
- Let $A \subseteq V_1$ arbitrary.
- $|\Gamma(A)| = |\{v \in V_2 \mid (v, w) \in E, w \in A\}|$.
- $|\Gamma(A)| \geq |\{v \in V_2 \mid (v, w) \in S, w \in A\}|$.
- $|\Gamma(A)| \geq \sum_{x_i \in A} d_i$. 
Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph with $V_1 = (x_1, ..., x_m)$ and $V_2 = (y_1, ..., y_n)$. Then contains $G$ a spanning graph $H$ with $\deg_H(x_i) = d_i$ and $0 \leq \deg_H(y_i) \leq 1$, iff for each $A \subseteq V_1$ we get

$$|\Gamma(A)| \geq \sum_{x_i \in A} d_i.$$

simple:

- Let $S$ be a spanning graph with $|S| = |V_1|$.
- Let $A \subset V_1$ arbitrary.
- $|\Gamma(A)| = |\{v \in V_2 \mid (v, w) \in E, w \in A\}|$.
- $|\Gamma(A)| \geq |\{v \in V_2 \mid (v, w) \in S, w \in A\}|$.
- $|\Gamma(A)| \geq \sum_{x_i \in A} d_i$. 
Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph with $V_1 = (x_1, ..., x_m)$ and $V_2 = (y_1, ..., y_n)$. Then contains $G$ a spanning graph $H$ with $\text{deg}_H(x_i) = d_i$ and $0 \leq \text{deg}_H(y_i) \leq 1$, iff for each $A \subseteq V_1$ we get

$$|\Gamma(A)| \geq \sum_{x_i \in A} d_i.$$
Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph with $V_1 = (x_1, \ldots, x_m)$ and $V_2 = (y_1, \ldots, y_n)$. Then contains $G$ a spanning graph $H$ with $\deg_H(x_i) = d_i$ and $0 \leq \deg_H(y_i) \leq 1$, iff for each $A \subseteq V_1$ we get

$$|\Gamma(A)| \geq \sum_{x_i \in A} d_i.$$
Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph with $V_1 = (x_1, \ldots, x_m)$ and $V_2 = (y_1, \ldots, y_n)$. Then contains $G$ a spanning graph $H$ with $\deg_H(x_i) = d_i$ and $0 \leq \deg_H(y_i) \leq 1$, iff for each $A \subseteq V_1$ we get

$$|\Gamma(A)| \geq \sum_{x_i \in A} d_i.$$

\[\implies\text{simple:}\]

- Let $S$ be a spanning graph with $|S| = |V_1|$.
- Let $A \subseteq V_1$ arbitrary.
- $|\Gamma(A)| = |\{v \in V_2 \mid (v, w) \in E, w \in A\}|$.
- $|\Gamma(A)| \geq |\{v \in V_2 \mid (v, w) \in S, w \in A\}|$.
- $|\Gamma(A)| \geq \sum_{x_i \in A} d_i$. 
Corollary

Let \( G = (V_1, V_2, E) \) be a bipartite graph with \( V_1 = (x_1, \ldots, x_m) \) and \( V_2 = (y_1, \ldots, y_n) \). Then contains \( G \) a spanning graph \( H \) with \( \deg_H(x_i) = d_i \) and \( 0 \leq \deg_H(y_i) \leq 1 \), iff for each \( A \subseteq V_1 \) we get

\[
|\Gamma(A)| \geq \sum_{x_i \in A} d_i.
\]

\( \rightleftharpoons \) by contradiction:

- Let \( S \) be the largest spanning graph with \( |S| < |V_1| \).
- Let \( A_1 = \{v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in S\} \).
- Let \( A_2 = \{v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in S\} \).
- Let \( a \in V_1 \setminus A_1 \).
- \( N(a) \cap A_2 \neq \emptyset \), because \( S \) is the largest spanning graph.
- Thus we get \( |\Gamma(A_1 \cup \{a\})| < \sum_{x_i \in A_1 \cup \{a\}} d_i \).
Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph with $V_1 = (x_1, \ldots, x_m)$ and $V_2 = (y_1, \ldots, y_n)$. Then contains $G$ a spanning graph $H$ with $\deg_H(x_i) = d_i$ and $0 \leq \deg_H(y_i) \leq 1$, iff for each $A \subseteq V_1$ we get

$$|\Gamma(A)| \geq \sum_{x_i \in A} d_i.$$ 

$\Leftarrow$ by contradiction:

- Let $S$ be the largest spanning graph with $|S| < |V_1|$.
- Let $A_1 = \{ v \in V_1 | \exists b \in V_2 : \{ v, b \} \in S \}$.
- Let $A_2 = \{ v \in V_2 | \exists b \in V_1 : \{ v, b \} \in S \}$.
- Let $a \in V_1 \setminus A_1$.
- $N(a) \cap A_2 \neq \emptyset$, because $S$ is the largest spanning graph.
- Thus we get $|\Gamma(A_1 \cup \{ a \})| < \sum_{x_i \in A_1 \cup \{ a \}} d_i$. 
Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph with $V_1 = (x_1, ..., x_m)$ and $V_2 = (y_1, ..., y_n)$. Then contains $G$ a spanning graph $H$ with $\deg_H(x_i) = d_i$ and $0 \leq \deg_H(y_i) \leq 1$, iff for each $A \subseteq V_1$ we get

$$|\Gamma(A)| \geq \sum_{x_i \in A} d_i.$$

\[\Leftarrow\text{ by contradiction:}\]

- Let $S$ be the largest spanning graph with $|S| < |V_1|$.
- Let $A_1 = \{v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in S\}$.
- Let $A_2 = \{v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in S\}$.
- Let $a \in V_1 \setminus A_1$.
- $N(a) \cap A_2 \neq \emptyset$, because $S$ is the largest spanning graph.
- Thus we get $|\Gamma(A_1 \cup \{a\})| < \sum_{x_i \in A_1 \cup \{a\}} d_i$. 


Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph with $V_1 = (x_1, \ldots, x_m)$ and $V_2 = (y_1, \ldots, y_n)$. Then contains $G$ a spanning graph $H$ with $\deg_H(x_i) = d_i$ and $0 \leq \deg_H(y_i) \leq 1$, iff for each $A \subseteq V_1$ we get

$$|\Gamma(A)| \geq \sum_{x_i \in A} d_i.$$  

by contradiction:

- Let $S$ be the largest spanning graph with $|S| < |V_1|$.
- Let $A_1 = \{ v \in V_1 \mid \exists b \in V_2 : \{ v, b \} \in S \}$.
- Let $A_2 = \{ v \in V_2 \mid \exists b \in V_1 : \{ v, b \} \in S \}$.
- Let $a \in V_1 \setminus A_1$.
- $N(a) \cap A_2 \neq \emptyset$, because $S$ is the largest spanning graph.
- Thus we get $|\Gamma(A_1 \cup \{ a \})| < \sum_{x_i \in A_1 \cup \{ a \}} d_i$. 
Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph with $V_1 = (x_1, \ldots, x_m)$ and $V_2 = (y_1, \ldots, y_n)$. Then contains $G$ a spanning graph $H$ with $\text{deg}_H(x_i) = d_i$ and $0 \leq \text{deg}_H(y_i) \leq 1$, iff for each $A \subseteq V_1$ we get

$$|\Gamma(A)| \geq \sum_{x_i \in A} d_i.$$

僖 by contradiction:

- Let $S$ be the largest spanning graph with $|S| < |V_1|$.
- Let $A_1 = \{v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in S\}$.
- Let $A_2 = \{v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in S\}$.
- Let $a \in V_1 \setminus A_1$.
- $N(a) \cap A_2 \neq \emptyset$, because $S$ is the largest spanning graph.
- Thus we get $|\Gamma(A_1 \cup \{a\})| < \sum_{x_i \in A_1 \cup \{a\}} d_i$. 
Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph with $V_1 = (x_1, \ldots, x_m)$ and $V_2 = (y_1, \ldots, y_n)$. Then contains $G$ a spanning graph $H$ with $\deg_H(x_i) = d_i$ and $0 \leq \deg_H(y_i) \leq 1$, iff for each $A \subseteq V_1$ we get

$$|\Gamma(A)| \geq \sum_{x_i \in A} d_i.$$

$\Leftarrow$ by contradiction:

- Let $S$ be the largest spanning graph with $|S| < |V_1|$.
- Let $A_1 = \{v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in S\}$.
- Let $A_2 = \{v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in S\}$.
- Let $a \in V_1 \setminus A_1$.
- $N(a) \cap A_2 \neq \emptyset$, because $S$ is the largest spanning graph.
- Thus we get $|\Gamma(A_1 \cup \{a\})| < \sum_{x_i \in A_1 \cup \{a\}} d_i$.  

Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph with $V_1 = (x_1, ..., x_m)$ and $V_2 = (y_1, ..., y_n)$. Then contains $G$ a spanning graph $H$ with $\text{deg}_H(x_i) = d_i$ and $0 \leq \text{deg}_H(y_i) \leq 1$, iff for each $A \subseteq V_1$ we get

$$|\Gamma(A)| \geq \sum_{x_i \in A} d_i.$$

\[\Leftarrow\text{ by contradiction:}\]

- Let $S$ be the largest spanning graph with $|S| < |V_1|$.
- Let $A_1 = \{v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in S\}$.
- Let $A_2 = \{v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in S\}$.
- Let $a \in V_1 \setminus A_1$.
- $N(a) \cap A_2 \neq \emptyset$, because $S$ is the largest spanning graph.
- Thus we get $|\Gamma(A_1 \cup \{a\})| < \sum_{x_i \in A_1 \cup \{a\}} d_i$. 
Corollary

Let \( G = (V_1, V_2, E) \) be a bipartite graph with \( V_1 = (x_1, \ldots, x_m) \) and \( V_2 = (y_1, \ldots, y_n) \). Then contains \( G \) a spanning graph \( H \) with \( \deg_H(x_i) = d_i \) and \( 0 \leq \deg_H(y_i) \leq 1 \), iff for each \( A \subseteq V_1 \) we get

\[
|\Gamma(A)| \geq \sum_{x_i \in A} d_i.
\]

\( \Leftarrow \) by contradiction:

- Let \( S \) be the largest spanning graph with \( |S| < |V_1| \).
- Let \( A_1 = \{v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in S\} \).
- Let \( A_2 = \{v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in S\} \).
- Let \( a \in V_1 \setminus A_1 \).
- \( N(a) \cap A_2 \neq \emptyset \), because \( S \) is the largest spanning graph.
- Thus we get \( |\Gamma(A_1 \cup \{a\})| < \sum_{x_i \in A_1 \cup \{a\}} d_i \).
Corollary

Let $G = (V_1, V_2, E)$ be a bipartite graph with $V_1 = (x_1, \ldots, x_m)$ and $V_2 = (y_1, \ldots, y_n)$. Then contains $G$ a spanning graph $H$ with $\deg_H(x_i) = d_i$ and $0 \leq \deg_H(y_i) \leq 1$, iff for each $A \subseteq V_1$ we get

$$|\Gamma(A)| \geq \sum_{x_i \in A} d_i.$$ 

← by contradiction:

- Let $S$ be the largest spanning graph with $|S| < |V_1|$.
- Let $A_1 = \{v \in V_1 \mid \exists b \in V_2 : \{v, b\} \in S\}$.
- Let $A_2 = \{v \in V_2 \mid \exists b \in V_1 : \{v, b\} \in S\}$.
- Let $a \in V_1 \setminus A_1$.
- $N(a) \cap A_2 \neq \emptyset$, because $S$ is the largest spanning graph.
- Thus we get $|\Gamma(A_1 \cup \{a\})| < \sum_{x_i \in A_1 \cup \{a\}} d_i$. 
Definition

Let $A = (a_{ij})$ be a matrix, $i = 1, ..., r$, $j = 1, ..., n$, with $a_{ij} \in \{1, ..., n\}$. The matrix $A$ is called Latin rectangle, iff no two element in a row or a column are the same.

Theorem

*Let $A$ be $r \times n$ Latin rectangle. Then we may enlarge $A$ to a $n \times n$ Latin square.*

Proof: Exercise.
**Definition**

Let $A = (a_{ij})$ be a matrix, $i = 1, \ldots, r$, $j = 1, \ldots, n$, with $a_{ij} \in \{1, \ldots, n\}$. The matrix $A$ is called Latin rectangle, iff no two element in a row or a column are the same.

**Theorem**

*Let $A$ be $r \times n$ Latin rectangle. Then we may enlarge $A$ to a $n \times n$ Latin square.*

Proof: Exercise.
Matching-Problems

**Definition (Maximal Matching Problem)**

Given: Graph $G = (V, E)$
Compute: Matching $M$ with: $\forall e \in E: M \cup \{e\}$ is no matching.

**Definition (Maximum Matching Problem)**

Given: Graph $G = (V, E)$
Compute: Matching $M$ with: $\forall M': M'$ is a matching $\implies |M'| \leq |M|$. 
Matching-Problems

**Definition (Maximal Matching Problem)**

Given: Graph $G = (V, E)$
Compute: Matching $M$ with: $\forall e \in E: M \cup \{e\}$ is no matching.

**Definition (Maximum Matching Problem)**

Given: Graph $G = (V, E)$
Compute: Matching $M$ with: $\forall M': M'$ is a matching $\implies |M'| \leq |M|$. 
The Maximal Matching Problem

Theorem (Maximal Matching Problem)

The maximal matching problem is in \( P \) for bipartite graphs.

Algorithm:

- Input: \( G = (V, E) \) bipartite graph.
- Let \( M = \emptyset \).
- While \( E \neq \emptyset \) do
  - Choose \( e \in E \)
  - Let \( M = M \cup \{e\} \)
  - Let \( E := E \setminus \{f \in E \mid e \cap f \neq \emptyset\} \)
The Maximal Matching Problem

Theorem (Maximal Matching Problem)

The maximal matching problem is in $\mathcal{P}$ for bipartite graphs.

Algorithm:

- **Input**: $G = (V, E)$ bipartite graph.
- Let $M = \emptyset$.
- While $E \neq \emptyset$ do
  - Choose $e \in E$
  - Let $M = M \cup \{e\}$
  - Let $E := E \setminus \{f \in E \mid e \cap f \neq \emptyset\}$
The Maximal Matching Problem

Theorem (Maximal Matching Problem)

*The maximal matching problem is in $\mathcal{P}$ for bipartite graphs.*

Algorithm:

- **Input:** $G = (V, E)$ bipartite graph.
- **Let** $M = \emptyset$.
- **While** $E \neq \emptyset$ do
  - Choose $e \in E$
  - Let $M = M \cup \{e\}$
  - Let $E := E \setminus \{f \in E \mid e \cap f \neq \emptyset\}$
The Maximal Matching Problem

Theorem (Maximal Matching Problem)

The maximal matching problem is in $\mathcal{P}$ for bipartite graphs.

Algorithm:

- **Input:** $G = (V, E)$ bipartite graph.
- Let $M = \emptyset$.

While $E \neq \emptyset$ do

- Choose $e \in E$
- Let $M = M \cup \{e\}$
- Let $E := E \setminus \{f \in E \mid e \cap f \neq \emptyset\}$
The Maximal Matching Problem

Theorem (Maximal Matching Problem)

The maximal matching problem is in \( \mathcal{P} \) for bipartite graphs.

Algorithm:

- **Input**: \( G = (V, E) \) bipartite graph.
- Let \( M = \emptyset \).
- **While** \( E \neq \emptyset \) do
  - Choose \( e \in E \)
  - Let \( M = M \cup \{e\} \)
  - Let \( E := E \setminus \{f \in E \mid e \cap f \neq \emptyset\} \)
The Maximal Matching Problem

Theorem (Maximal Matching Problem)

*The maximal matching problem is in $\mathcal{P}$ for bipartite graphs.*

Algorithm:

- **Input:** $G = (V, E)$ bipartite graph.
- Let $M = \emptyset$.
- While $E \neq \emptyset$ do
  - Choose $e \in E$
  - Let $M = M \cup \{e\}$
  - Let $E := E \setminus \{f \in E \mid e \cap f \neq \emptyset\}$
The Maximal Matching Problem

Theorem (Maximal Matching Problem)

The maximal matching problem is in $\mathcal{P}$ for bipartite graphs.

Algorithm:

- **Input:** $G = (V, E)$ bipartite graph.
- Let $M = \emptyset$.
- While $E \neq \emptyset$ do
  - Choose $e \in E$
  - Let $M = M \cup \{e\}$
  - Let $E := E \setminus \{f \in E \mid e \cap f \neq \emptyset\}$
The Maximal Matching Problem

**Theorem (Maximal Matching Problem)**

The maximal matching problem is in $\mathcal{P}$ for bipartite graphs.

**Algorithm:**

- **Input:** $G = (V, E)$ bipartite graph.
- **Let** $M = \emptyset$.
- **While** $E \neq \emptyset$ do
  - Choose $e \in E$
  - **Let** $M = M \cup \{e\}$
  - **Let** $E := E \setminus \{f \in E \mid e \cap f \neq \emptyset\}$
The Maximal Matching Problem

Theorem (Maximal Matching Problem)

The maximal matching problem is in $\mathcal{P}$ for bipartite graphs.

Algorithm:

- **Input**: $G = (V, E)$ bipartite graph.
- **Let** $M = \emptyset$.
- **While** $E \neq \emptyset$ do
  - **Choose** $e \in E$
  - **Let** $M = M \cup \{e\}$
  - **Let** $E := E \setminus \{f \in E \mid e \cap f \neq \emptyset\}$
Alternating Paths

- Let $G = (V, E)$ be a graph and $M \subset E$ be a matching.

- A node $v \in V$ is called free, iff $v \notin \cup_{e \in M} e$.

- A path $v_0, \{v_0, v_1\}, v_1, \{v_1, v_2\}, v_2, \{v_2, v_3\}, v_3, \ldots, v_{l-1}, \{v_{l-1}, v_l\}, v_l$ is called alternating, iff $\{v_{i-1}, v_i\} \in M \iff \{v_i, v_{i+1}\} \notin M$ ($0 < i < l$).

- A alternating path $v_0, \{v_0, v_1\}, v_1, \{v_1, v_2\}, v_2, \{v_2, v_3\}, v_3, \ldots, v_{l-1}, \{v_{l-1}, v_l\}, v_l$ is called enlarging, iff $v_0, v_l$ are free.

- Note: An edge between free nodes is an enlarging path.

- We get the following algorithm:
  1. Let $M = \emptyset$.
  2. While there is an enlarging path $P$, do:
     1. Enlarge $M$ by the following operation $M = M \oplus E(P)$.
Alternating Paths

- Let $G = (V, E)$ be a graph and $M \subset E$ be a matching.
- A node $v \in V$ is called free, iff $v \notin \bigcup_{e \in M} e$.
- A path $v_0, \{v_0, v_1\}, v_1, \{v_1, v_2\}, v_2, \{v_2, v_3\}, v_3, \ldots, v_{i-1}, \{v_{i-1}, v_i\}, v_i$ is called alternating, iff $\{v_{i-1}, v_i\} \in M \iff \{v_i, v_{i+1}\} \notin M$ ($0 < i < l$).

- A alternating path
  $v_0, \{v_0, v_1\}, v_1, \{v_1, v_2\}, v_2, \{v_2, v_3\}, v_3, \ldots, v_{i-1}, \{v_{i-1}, v_i\}, v_i$ is called enlarging, iff $v_0, v_i$ are free.

- Note: An edge between free nodes is an enlarging path.

- We get the following algorithm:
  1. Let $M = \emptyset$.
  2. While there is an enlarging path $P$, do:
     1. Enlarge $M$ by the following operation $M = M \oplus E(P)$. 

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]
Alternating Paths

- Let $G = (V, E)$ be a graph and $M \subset E$ be a matching.
- A node $v \in V$ is called free, iff $v \notin \bigcup_{e \in M} e$.
- A path $v_0, \{v_0, v_1\}, v_1, \{v_1, v_2\}, v_2, \{v_2, v_3\}, v_3, \ldots, v_{i-1}, \{v_{i-1}, v_i\}, v_i$ is called alternating, iff $\{v_{i-1}, v_i\} \in M \iff \{v_i, v_{i+1}\} \notin M$ (0 < $i$ < $l$).

A alternating path

$v_0, \{v_0, v_1\}, v_1, \{v_1, v_2\}, v_2, \{v_2, v_3\}, v_3, \ldots, v_{i-1}, \{v_{i-1}, v_i\}, v_i$ is called enlarging, iff $v_0, v_l$ are free.

Note: An edge between free nodes is an enlarging path.

We get the following algorithm:

1. Let $M = \emptyset$.
2. While there is an enlarging path $P$, do:
   1. Enlarge $M$ by the following operation $M = M \oplus E(P)$. 

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]
Let $G = (V, E)$ be a graph and $M \subset E$ be a matching.

A node $v \in V$ is called free, iff $v \notin \bigcup_{e \in M} e$.

A path $v_0, \{v_0, v_1\}, v_1, \{v_1, v_2\}, v_2, \{v_2, v_3\}, v_3, \ldots, v_{i-1}, \{v_{i-1}, v_i\}, v_i$ is called alternating, iff $\{v_{i-1}, v_i\} \in M \Leftrightarrow \{v_i, v_{i+1}\} \notin M$ ($0 < i < I$).

A alternating path
$v_0, \{v_0, v_1\}, v_1, \{v_1, v_2\}, v_2, \{v_2, v_3\}, v_3, \ldots, v_{i-1}, \{v_{i-1}, v_i\}, v_i$ is called enlarging, iff $v_0, v_i$ are free.

Note: An edge between free nodes is an enlarging path.

We get the following algorithm:

1. Let $M = \emptyset$.
2. While there is an enlarging path $P$, do:
   1. Enlarge $M$ by the following operation $M = M \oplus E(P)$.
Alternating Paths

- Let $G = (V, E)$ be a graph and $M \subset E$ be a matching.
- A node $v \in V$ is called free, iff $v \notin \bigcup_{e \in M} e$.
- A path $v_0, \{v_0, v_1\}, v_1, \{v_1, v_2\}, v_2, \{v_2, v_3\}, v_3, \ldots, v_{l-1}, \{v_{l-1}, v_l\}, v_l$ is called alternating, iff $\{v_{i-1}, v_i\} \in M \iff \{v_i, v_{i+1}\} \notin M$ ($0 < i < l$).
- An alternating path $v_0, \{v_0, v_1\}, v_1, \{v_1, v_2\}, v_2, \{v_2, v_3\}, v_3, \ldots, v_{l-1}, \{v_{l-1}, v_l\}, v_l$ is called enlarging, iff $v_0, v_l$ are free.
- Note: An edge between free nodes is an enlarging path.
- We get the following algorithm:
  1. Let $M = \emptyset$.
  2. While there is an enlarging path $P$, do:
     1. Enlarge $M$ by the following operation $M = M \oplus E(P)$. 

$A \oplus B = (A \cup B) \setminus (A \cap B)$
Let $G = (V, E)$ be a graph and $M \subset E$ be a matching.

A node $v \in V$ is called free, iff $v \notin \bigcup_{e \in M} e$.

A path $v_0, \{v_0, v_1\}, v_1, \{v_1, v_2\}, v_2, \{v_2, v_3\}, v_3, \ldots, v_{i-1}, \{v_{i-1}, v_i\}, v_i$ is called alternating, iff $\{v_{i-1}, v_i\} \in M \iff \{v_i, v_{i+1}\} \notin M$ ($0 < i < l$).

A alternating path $v_0, \{v_0, v_1\}, v_1, \{v_1, v_2\}, v_2, \{v_2, v_3\}, v_3, \ldots, v_{i-1}, \{v_{i-1}, v_i\}, v_i$ is called enlarging, iff $v_0, v_i$ are free.

Note: An edge between free nodes is an enlarging path.

We get the following algorithm:

1. Let $M = \emptyset$.
2. While there is an enlarging path $P$, do:
   1. Enlarge $M$ by the following operation $M = M \oplus E(P)$. 

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]
Alternating Paths

- Let $G = (V, E)$ be a graph and $M \subset E$ be a matching.

- A node $v \in V$ is called free, iff $v \notin \bigcup_{e \in M} e$.

- A path $v_0, \{v_0, v_1\}, v_1, \{v_1, v_2\}, v_2, \{v_2, v_3\}, v_3, \ldots, v_{l-1}, \{v_{l-1}, v_l\}, v_l$ is called alternating, iff $\{v_{i-1}, v_i\} \in M \Leftrightarrow \{v_i, v_{i+1}\} \notin M$ ($0 < i < l$).

- A alternating path $v_0, \{v_0, v_1\}, v_1, \{v_1, v_2\}, v_2, \{v_2, v_3\}, v_3, \ldots, v_{l-1}, \{v_{l-1}, v_l\}, v_l$ is called enlarging, iff $v_0, v_l$ are free.

- Note: An edge between free nodes is an enlarging path.

- We get the following algorithm:

  1. Let $M = \emptyset$.
  2. While there is an enlarging path $P$, do:
     1. Enlarge $M$ by the following operation $M = M \oplus E(P)$. 

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]
Let $G = (V, E)$ be a graph and $M \subset E$ be a matching.

A node $v \in V$ is called free, iff $v \not\in \bigcup_{e \in M} e$.

A path $v_0, \{v_0, v_1\}, v_1, \{v_1, v_2\}, v_2, \{v_2, v_3\}, v_3, \ldots, v_{l-1}, \{v_{l-1}, v_l\}, v_l$ is called alternating, iff $\{v_{i-1}, v_i\} \in M \iff \{v_i, v_{i+1}\} \not\in M \ (0 < i < l)$.

A alternating path $v_0, \{v_0, v_1\}, v_1, \{v_1, v_2\}, v_2, \{v_2, v_3\}, v_3, \ldots, v_{l-1}, \{v_{l-1}, v_l\}, v_l$ is called enlarging, iff $v_0, v_l$ are free.

Note: An edge between free nodes is an enlarging path.

We get the following algorithm:

1. Let $M = \emptyset$.
2. While there is an enlarging path $P$, do:
   1. Enlarge $M$ by the following operation $M = M \oplus E(P)$. 
Alternating Paths

- Let $G = (V, E)$ be a graph and $M \subset E$ be a matching.

- A node $v \in V$ is called free, iff $v \not\in \bigcup_{e \in M} e$.

- A path $v_0, \{v_0, v_1\}, v_1, \{v_1, v_2\}, v_2, \{v_2, v_3\}, v_3, \ldots, v_{l-1}, \{v_{l-1}, v_l\}, v_l$ is called alternating, iff $\{v_{i-1}, v_i\} \in M \iff \{v_i, v_{i+1}\} \not\in M$ ($0 < i < l$).

- A alternating path $v_0, \{v_0, v_1\}, v_1, \{v_1, v_2\}, v_2, \{v_2, v_3\}, v_3, \ldots, v_{l-1}, \{v_{l-1}, v_l\}, v_l$ is called enlarging, iff $v_0, v_l$ are free.

- Note: An edge between free nodes is an enlarging path.

- We get the following algorithm:

  1. Let $M = \emptyset$.
  2. While there is an enlarging path $P$, do:
     1. Enlarge $M$ by the following operation $M = M \oplus E(P)$.
Alternating Paths

Let $G = (V, E)$ be a graph and $M \subset E$ be a matching.

A node $v \in V$ is called free, iff $v \notin \bigcup_{e \in M} e$.

A path $v_0, \{v_0, v_1\}, v_1, \{v_1, v_2\}, v_2, \{v_2, v_3\}, v_3, \ldots, v_{l-1}, \{v_{l-1}, v_l\}, v_l$ is called alternating, iff $\{v_{i-1}, v_i\} \in M \Leftrightarrow \{v_i, v_{i+1}\} \notin M$ ($0 < i < l$).

A alternating path $v_0, \{v_0, v_1\}, v_1, \{v_1, v_2\}, v_2, \{v_2, v_3\}, v_3, \ldots, v_{l-1}, \{v_{l-1}, v_l\}, v_l$ is called enlarging, iff $v_0, v_l$ are free.

Note: An edge between free nodes is an enlarging path.

We get the following algorithm:

1. Let $M = \emptyset$.
2. While there is an enlarging path $P$, do:
   1. Enlarge $M$ by the following operation $M = M \oplus E(P)$.
Try enlarging paths on arbitrary graphs:

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]
Example arbitrary Graph

Try enlarging paths on arbitrary graphs:

\( A \oplus B = (A \cup B) \setminus (A \cap B) \)
Example arbitrary Graph

Try enlarging paths on arbitrary graphs:

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]
Example arbitrary Graph

Try enlarging paths on arbitrary graphs:

\[
A \oplus B = (A \cup B) \setminus (A \cap B)
\]
Example arbitrary Graph

Try enlarging paths on arbitrary graphs:

$$A \oplus B = (A \cup B) \setminus (A \cap B)$$
Example arbitrary Graph

Try enlarging paths on arbitrary graphs:

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]
Example arbitrary Graph

Try enlarging paths on arbitrary graphs:

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]
Example arbitrary Graph

Try enlarging paths on arbitrary graphs:

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]
Example arbitrary Graph

Try enlarging paths on arbitrary graphs:

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]
Example arbitrary Graph

Try enlarging paths on arbitrary graphs:

A ⊕ B = (A ∪ B) \ (A ∩ B)
Example arbitrary Graph

Try enlarging paths on arbitrary graphs:

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]
Example arbitrary Graph

Try enlarging paths on arbitrary graphs:

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]
Basic Definitions
Connectivity of Graphs
Flows
Matchings
Factors of Graphs
Posets

Probleme (1:40.13)

Example arbitrary Graph

Try enlarging paths on arbitrary graphs:

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]
Example arbitrary Graph

Try enlarging paths on arbitrary graphs:

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]
Example arbitrary Graph

Try enlarging paths on arbitrary graphs:

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]
Example arbitrary Graph

Try enlarging paths on arbitrary graphs:

\[
A \oplus B = (A \cup B) \setminus (A \cap B)
\]
Example arbitrary Graph

Try enlarging paths on arbitrary graphs:

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]
Example arbitrary Graph

Try enlarging paths on arbitrary graphs:

Odd cycles could become a problem

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]
Example arbitrary Graph

Try enlarging paths on arbitrary graphs:

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]

Odd cycles could become a problem
Example arbitrary Graph

Try enlarging paths on arbitrary graphs:

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]

Odd cycles could become a problem
Example arbitrary Graph

Try enlarging paths on arbitrary graphs:

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]

Odd cycles could become a problem
Example arbitrary Graph

Try enlarging paths on arbitrary graphs:

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]

Odd cycles could become a problem
Try enlarging paths on arbitrary graphs:

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]

Odd cycles could become a problem
Theorem of Berge

**Theorem (Berge)**

A matching $M'$ of a graph $G$ is a maximum Matching, iff there exists no enlarging path.

**Proof:**

$\implies$ simple.

$\iff$ by contradiction.

- Let $M$ be a matching with $|M| > |M'|$ and assume there is no enlarging path for $M'$.
- Consider the graph $H$ containing only edges from $M \cup M' \setminus (M \cap M')$.
- $H$ consists of disjoint paths and cycles.
- Thus there is a enlarging path $M'$. 

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]
Theorem of Berge

**Theorem (Berge)**

A matching $M'$ of a graph $G$ is a maximum Matching, iff there exists no enlarging path.

**Proof:**

$\implies$ simple.

$\impliedby$ by contradiction.

- Let $M$ be a matching with $|M| > |M'|$ and assume there is no enlarging path for $M'$.
- Consider the graph $H$ containing only edges from $M \cup M' \setminus (M \cap M')$.
- $H$ consists of disjoint paths and cycles.
- Thus there is a enlarging path $M'$. 

$A \oplus B = (A \cup B) \setminus (A \cap B)$
**Theorem of Berge**

A matching $M'$ of a graph $G$ is a maximum Matching, iff there exists no enlarging path.

**Proof:**

$\implies$ simple.

$\iff$ by contradiction.

- Let $M$ be a matching with $|M| > |M'|$ and assume there is no enlarging path for $M'$.
- Consider the graph $H$ containing only edges from $M \cup M' \setminus (M \cap M')$.
- $H$ consists of disjoint paths and cycles.
- Thus there is a enlarging path $M'$. 

$$A \oplus B = (A \cup B) \setminus (A \cap B)$$
**Theorem of Berge**

A matching $M'$ of a graph $G$ is a maximum Matching, iff there exists no enlarging path.

**Proof:**

$\implies$ simple.

$\Leftarrow$ by contradiction.

- Let $M$ be a matching with $|M| > |M'|$ and assume there is no enlarging path for $M'$.
- Consider the graph $H$ containing only edges from $M \cup M' \setminus (M \cap M')$.
- $H$ consists of disjoint paths and cycles.
- Thus there is a enlarging path $M'$. 

$A \oplus B = (A \cup B) \setminus (A \cap B)$
Theorem of Berge

A matching $M'$ of a graph $G$ is a maximum Matching, iff there exists no enlarging path.

Proof:

$\implies$ simple.

$\impliedby$ by contradiction.

- Let $M$ be a matching with $|M| > |M'|$ and assume there is no enlarging path for $M'$.
- Consider the graph $H$ containing only edges from $M \cup M' \setminus (M \cap M')$.
- $H$ consists of disjoint paths and cycles.
- Thus there is an enlarging path $M'$. 

$A \oplus B = (A \cup B) \setminus (A \cap B)$
Theorem of Berge

A matching $M'$ of a graph $G$ is a maximum Matching, iff there exists no enlarging path.

Proof:

$\implies$ simple.

$\impliedby$ by contradiction.

- Let $M$ be a matching with $|M| > |M'|$ and assume there is no enlarging path for $M'$.
- Consider the graph $H$ containing only edges from $M \cup M' \setminus (M \cap M')$.
- $H$ consists of disjoint paths and cycles.
- Thus there is a enlarging path $M'$. 

$A \oplus B = (A \cup B) \setminus (A \cap B)$
Theorem of Berge

Theorem (Berge)

A matching $M'$ of a graph $G$ is a maximum Matching, iff there exists no enlarging path.

Proof:

$\implies$ simple.

$\iff$ by contradiction.

- Let $M$ be a matching with $|M| > |M'|$ and assume there is no enlarging path for $M'$.
- Consider the graph $H$ containing only edges from $M \cup M' \setminus (M \cap M')$.
- $H$ consists of disjoint paths and cycles.
- Thus there is a enlarging path $M'$. 

$$A \oplus B = (A \cup B) \setminus (A \cap B)$$
Theorem of Berge

**Theorem (Berge)**

*A matching \( M' \) of a graph \( G \) is a maximum Matching, iff there exists no enlarging path.*

**Proof:**

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]

\[ \implies \text{ simple.} \]

\[ \iff \text{ by contradiction.} \]

- Let \( M \) be a matching with \( |M| > |M'| \) and assume there is no enlarging path for \( M' \).
- Consider the graph \( H \) containing only edges from \( M \cup M' \setminus (M \cap M') \).
- \( H \) consists of disjoint paths and cycles.
- Thus there is a enlarging path \( M' \).
The Maximum Matching Problem

Theorem (Maximum Matching Problem)

The Maximum Matching Problem ist in $P$.

Algorithm:

- Input $G = (V, E)$ [bipartite] graph.
- Let $M = \emptyset$.
- While there is an enlarging path $(a_0, a_1, a_2, \cdots a_l)$ in $G$, with odd $l$, $\{a_{2i}, a_{2i+1}\} \not\in M$ and $\{a_{2i+1}, a_{2i}\} \in M$ do
  - Exchange the edges of $P$:
    - Add the edges of the form $\{a_{2i}, a_{2i+1}\}$ to $M$ and
    - delete the edges of the form $\{a_{2i+1}, a_{2i}\}$ from $M$.
- If $G = (V, E)$ is not bipartite graph, then resolve the odd cycles recursively.
The Maximum Matching Problem

Algorithm:

- **Input** \( G = (V, E) \) [bipartite] graph.
- Let \( M = \emptyset \).
- While there is an enlarging path \( (a_0, a_1, a_2, \cdots a_l) \) in \( G \), with odd \( l \), \( \{a_{2i}, a_{2i+1}\} \not\in M \) and \( \{a_{2i+1}, a_{2i}\} \in M \) do
  - Exchange the edges of \( P \):
    - Add the edges of the form \( \{a_{2i}, a_{2i+1}\} \) to \( M \) and
    - delete the edges of the form \( \{a_{2i+1}, a_{2i}\} \) from \( M \).
- If \( G = (V, E) \) is not bipartite graph, then resolve the odd cycles recursively.
The Maximum Matching Problem

Theorem (Maximum Matching Problem)

The Maximum Matching Problem is in \( \mathcal{P} \).

Algorithm:

- **Input** \( G = (V, E) \) [bipartite] graph.
- **Let** \( M = \emptyset \).

- While there is an enlarging path \( (a_0, a_1, a_2, \cdots a_l) \) in \( G \), with odd \( l \), \( \{a_{2 \cdot i}, a_{2 \cdot i+1}\} \not\in M \) and \( \{a_{2 \cdot i+1}, a_{2 \cdot i}\} \in M \) do
  - Exchange the edges of \( P \):
    - Add the edges of the form \( \{a_{2 \cdot i}, a_{2 \cdot i+1}\} \) to \( M \) and
    - delete the edges of the form \( \{a_{2 \cdot i+1}, a_{2 \cdot i}\} \) from \( M \).

- If \( G = (V, E) \) is not bipartite graph, then resolve the odd cycles recursively.
The Maximum Matching Problem

Theorem (Maximum Matching Problem)

The Maximum Matching Problem is in \( \mathcal{P} \).

Algorithm:

- Input \( G = (V, E) \) [bipartite] graph.
- Let \( M = \emptyset \).
- While there is an enlarging path \((a_0, a_1, a_2, \cdots, a_l)\) in \( G \), with odd \( l \), \( \{a_{2i}, a_{2i+1}\} \not\in M \) and \( \{a_{2i+1}, a_{2i}\} \in M \) do
  - Exchange the edges of \( P \):
    - Add the edges of the form \( \{a_{2i}, a_{2i+1}\} \) to \( M \) and
    - delete the edges of the form \( \{a_{2i+1}, a_{2i}\} \) from \( M \).
- If \( G = (V, E) \) is not bipartite graph, then resolve the odd cycles recursively.
The Maximum Matching Problem

**Theorem (Maximum Matching Problem)**

*The Maximum Matching Problem ist in $\mathcal{P}$.*

**Algorithm:**

- **Input** $G = (V, E)$ [bipartite] graph.
- **Let** $M = \emptyset$.
- **While** there is an enlarging path $(a_0, a_1, a_2, \cdots a_l)$ in $G$, with odd $l$, 
  $\{a_{2i}, a_{2i+1}\} \not\in M$ and $\{a_{2i+1}, a_{2i}\} \in M$ do
  - **Exchange the edges of $P$:**
    - Add the edges of the form $\{a_{2i}, a_{2i+1}\}$ to $M$ and
    - delete the edges of the form $\{a_{2i+1}, a_{2i}\}$ from $M$.
- **If** $G = (V, E)$ is not bipartite graph, then resolve the odd cycles recursively.
The Maximum Matching Problem

**Theorem (Maximum Matching Problem)**

*The Maximum Matching Problem ist in $\mathcal{P}$.***

Algorithm:

- **Input** $G = (V, E)$ [bipartite] graph.
- **Let** $M = \emptyset$.
- **While** there is an enlarging path $(a_0, a_1, a_2, \cdots a_l)$ in $G$, with odd $l$, 
  $\{a_{2i}, a_{2i+1}\} \notin M$ and $\{a_{2i+1}, a_{2i}\} \in M$ do
  - Exchange the edges of $P$:
    - **Add** the edges of the form $\{a_{2i}, a_{2i+1}\}$ to $M$ and
    - **delete** the edges of the form $\{a_{2i+1}, a_{2i}\}$ from $M$.
- **If** $G = (V, E)$ is not bipartite graph, then resolve the odd cycles recursively.
The Maximum Matching Problem

**Theorem (Maximum Matching Problem)**

The Maximum Matching Problem ist in \( \mathcal{P} \).

**Algorithm:**

- Input \( G = (V, E) \) [bipartite] graph.
- Let \( M = \emptyset \).
- While there is an enlarging path \( (a_0, a_1, a_2, \cdots a_l) \) in \( G \), with odd \( l \), \( \{a_2i, a_2i+1\} \not\in M \) and \( \{a_2i+1, a_2i\} \in M \) do
  - Exchange the edges of \( P \):
    - Add the edges of the form \( \{a_2i, a_2i+1\} \) to \( M \) and
    - delete the edges of the form \( \{a_2i+1, a_2i\} \) from \( M \).
- If \( G = (V, E) \) is not bipartite graph, then resolve the odd cycles recursively.
The Maximum Matching Problem

Theorem (Maximum Matching Problem)

The Maximum Matching Problem ist in \( \mathcal{P} \).

Algorithm:

- Input \( G = (V, E) \) [bipartite] graph.
- Let \( M = \emptyset \).
- While there is an enlarging path \((a_0, a_1, a_2, \cdots a_l)\) in \( G \), with odd \( l \), \( \{a_{2i}, a_{2i+1}\} \not\in M \) and \( \{a_{2i+1}, a_{2i}\} \in M \) do
  - Exchange the edges of \( P \):
    - Add the edges of the form \( \{a_{2i}, a_{2i+1}\} \) to \( M \) and
    - delete the edges of the form \( \{a_{2i+1}, a_{2i}\} \) from \( M \).
- If \( G = (V, E) \) is not bipartite graph, then resolve the odd cycles recursively.
The Maximum Matching Problem

Theorem (Maximum Matching Problem)

\[ A \oplus B = (A \cup B) \setminus (A \cap B) \]

The Maximum Matching Problem is in \( \mathcal{P} \).

Algorithm:

- Input \( G = (V, E) \) [bipartite] graph.
- Let \( M = \emptyset \).
- While there is an enlarging path \( (a_0, a_1, a_2, \ldots, a_l) \) in \( G \), with odd \( l \), \( \{a_{2i}, a_{2i+1}\} \not\in M \) and \( \{a_{2i+1}, a_{2i}\} \in M \) do
  - Exchange the edges of \( P \):
    - Add the edges of the form \( \{a_{2i}, a_{2i+1}\} \) to \( M \) and
    - delete the edges of the form \( \{a_{2i+1}, a_{2i}\} \) from \( M \).
- If \( G = (V, E) \) is not bipartite graph, then resolve the odd cycles recursively.
Factors

**Definition**
Let $G$ be a graph. A $k$-regular spanning graph $H$ of $G$ is called $k$-factor.

**Theorem**
The graph $K_{2t}$ is the sum of $2t - 1$ 1-factors.

**Theorem**
The graph $K_{2t+1}$ is the sum of $t$ spanning cycles.
Factors

Definition
Let $G$ be a graph. A $k$-regular spanning graph $H$ of $G$ is called $k$-factor.

Theorem
The graph $K_{2t}$ is the sum of $2t - 1$ 1-factors.

Theorem
The graph $K_{2t+1}$ is the sum of $t$ spanning cycles.
Factors

**Definition**
Let $G$ be a graph. A $k$-regular spanning graph $H$ of $G$ is called $k$-factor.

**Theorem**
The graph $K_{2t}$ is the sum of $2t - 1$ 1-factors.

**Theorem**
The graph $K_{2t+1}$ is the sum of $t$ spanning cycles.
Example I
Example I
Example I
Example 1
Example I

\[ a_1 \rightarrow a_2 \rightarrow a_3 \rightarrow a_4 \rightarrow a_5 \rightarrow a_6 \rightarrow a_7 \rightarrow a_8 \rightarrow a_9 \rightarrow a_{10} \]
Example I
Example 1
Example I
Example 1
Example 1
**Theorem**

The graph $K_{2t}$ is the sum of $2t - 1$ 1-factors.

- Draw $2t - 1$ nodes $a_1, a_2, \ldots, a_{2t-1}$, as a regular $(2t - 1)$-gon.
- Draw $a_{2t}$ as the top of a pyramid above the nodes $a_1, a_2, \ldots, a_{2t-1}$.
- Choose a 1-Factor:
  - Choose one edge of the $(2t - 1)$-gon.
  - Choose all parallel diagonals in the $(2t - 1)$-gon.
  - Choose one edge from the only free node of the $(2t - 1)$-gon to the top.
Proof I

**Theorem**

The graph $K_{2t}$ is the sum of $2t - 1$ 1-factors.

- Draw $2t - 1$ nodes $a_1, a_2, \ldots, a_{2t-1}$, as a regular $(2t - 1)$-gon.
- Draw $a_{2t}$ as the top of a pyramid above the nodes $a_1, a_2, \ldots, a_{2t-1}$.
- Choose a 1-Factor:
  - Choose one edge of the $(2t - 1)$-gon.
  - Choose all parallel diagonals in the $(2t - 1)$-gon.
  - Choose one edge from the only free node of the $(2t - 1)$-gon to the top.
**Proof I**

**Theorem**

The graph $K_{2t}$ is the sum of $2t - 1$ 1-factors.

- Draw $2t - 1$ nodes $a_1, a_2, \ldots, a_{2t-1}$, as a regular $(2t - 1)$-gon.
- Draw $a_{2t}$ as the top of a pyramid above the nodes $a_1, a_2, \ldots, a_{2t-1}$.
- **Choose a 1-Factor:**
  - Choose one edge of the $(2t - 1)$-gon.
  - Choose all parallel diagonals in the $(2t - 1)$-gon.
  - Choose one edge from the only free node of the $(2t - 1)$-gon to the top.
**Theorem**

The graph $K_{2t}$ is the sum of $2t - 1$ 1-factors.

- Draw $2t - 1$ nodes $a_1, a_2, \ldots, a_{2t-1}$, as a regular $(2t - 1)$-gon.
- Draw $a_{2t}$ as the top of a pyramid above the nodes $a_1, a_2, \ldots, a_{2t-1}$.
- Choose a 1-Factor:
  - Choose one edge of the $(2t - 1)$-gon.
  - Choose all parallel diagonals in the $(2t - 1)$-gon.
  - Choose one edge from the only free node of the $(2t - 1)$-gon to the top.
Proof I

Theorem

The graph $K_{2t}$ is the sum of $2t - 1$ 1-factors.

- Draw $2t - 1$ nodes $a_1, a_2, \cdots, a_{2t-1}$, as a regular $(2t - 1)$-gon.
- Draw $a_{2t}$ as the top of a pyramid above the nodes $a_1, a_2, \cdots, a_{2t-1}$.
- Choose a 1-Factor:
  - Choose one edge of the $(2t - 1)$-gon.
  - Choose all parallel diagonals in the $(2t - 1)$-gon.
  - Choose one edge from the only free node of the $(2t - 1)$-gon to the top.
Theorem

The graph $K_{2t}$ is the sum of $2t - 1$ 1-factors.

- Draw $2t - 1$ nodes $a_1, a_2, \ldots, a_{2t-1}$, as a regular $(2t - 1)$-gon.
- Draw $a_{2t}$ as the top of a pyramid above the nodes $a_1, a_2, \ldots, a_{2t-1}$.
- Choose a 1-Factor:
  - Choose one edge of the $(2t - 1)$-gon.
  - Choose all parallel diagonals in the $(2t - 1)$-gon.
  - Choose one edge from the only free node of the $(2t - 1)$-gon to the top.
The graph $K_{2t}$ is the sum of $2t - 1$ 1-factors.

- Draw $2t - 1$ nodes $a_1, a_2, \cdots, a_{2t-1}$, as a regular $(2t - 1)$-gon.
- Draw $a_{2t}$ as the top of a pyramid above the nodes $a_1, a_2, \cdots, a_{2t-1}$.
- Choose a 1-Factor:
  - Choose one edge of the $(2t - 1)$-gon.
  - Choose all parallel diagonals in the $(2t - 1)$-gon.
  - Choose one edge from the only free node of the $(2t - 1)$-gon to the top.
Example II
Example II
Example II
Example II
Example II
Example II
Example II
Example II
Example II
Example II
Proof II

Theorem

*The graph $K_{2t+1}$ is the sum of $t$ spanning cycles.*

- **Draw** $2t$ nodes $a_1, a_2, \ldots, a_{2t}$, as a regular $(2t)$-gon.
- **Draw** $a_{2t+1}$ as the top of a pyramid above the nodes $a_1, a_2, \ldots, a_{2t}$.
- **Choose one 2-factor:**
  - connect two opposing nodes as follows:
  - Move in a zig-zag way over all nodes of the $(2t)$-gon.
  - Move first to the direct right neighbour,
  - and then to the direct left neighbour (i.e. two nodes back).
  - Continue in the same fashion.
  - Connect the two opposing end-nodes through $a_{2t+1}$

- We may identify for each edge a unique 2-factor.
The graph $K_{2t+1}$ is the sum of $t$ spanning cycles.

- Draw $2t$ nodes $a_1, a_2, \ldots, a_{2t}$, as a regular $(2t)$-gon.
- Draw $a_{2t+1}$ as the top of a pyramid above the nodes $a_1, a_2, \ldots, a_{2t}$.
- Choose one 2-factor:
  - connect two opposing nodes as follows:
  - Move in a zig-zag way over all nodes of the $(2t)$-gon.
  - Move first to the direct right neighbour,
  - and then to the direct left neighbour (i.e. two nodes back).
  - Continue in the same fashion.
  - Connect the two opposing end-nodes through $a_{2t+1}$
- We may identify for each edge a unique 2-factor.
Proof II

**Theorem**

*The graph $K_{2t+1}$ is the sum of $t$ spanning cycles.*

- Draw $2t$ nodes $a_1, a_2, \cdots, a_{2t}$, as a regular $(2t)$-gon.
- Draw $a_{2t+1}$ as the top of a pyramid above the nodes $a_1, a_2, \cdots, a_{2t}$.
- **Choose one 2-factor:**
  - connect two opposing nodes as follows:
  - Move in a zig-zag way over all nodes of the $(2t)$-gon.
  - Move first to the direct right neighbour,
  - and then to the direct left neighbour (i.e. two nodes back).
  - Continue in the same fashion.
  - Connect the two opposing end-nodes through $a_{2t+1}$
- We may identify for each edge a unique 2-factor.
Proof II

**Theorem**

*The graph $K_{2t+1}$ is the sum of $t$ spanning cycles.*

- Draw $2t$ nodes $a_1, a_2, \ldots, a_{2t}$, as a regular $(2t)$-gon.
- Draw $a_{2t+1}$ as the top of a pyramid above the nodes $a_1, a_2, \ldots, a_{2t}$.
- Choose one 2-factor:
  - connect two opposing nodes as follows:
    - Move in a zig-zag way over all nodes of the $(2t)$-gon.
    - Move first to the direct right neighbour,
    - and then to the direct left neighbour (i.e. two nodes back).
    - Continue in the same fashion.
    - Connect the two opposing end-nodes through $a_{2t+1}$
- We may identify for each edge a unique 2-factor.
Theorem

The graph $K_{2t+1}$ is the sum of $t$ spanning cycles.

- Draw $2t$ nodes $a_1, a_2, \ldots, a_{2t}$, as a regular $(2t)$-gon.
- Draw $a_{2t+1}$ as the top of a pyramid above the nodes $a_1, a_2, \ldots, a_{2t}$.
- Choose one 2-factor:
  - connect two opposing nodes as follows:
  - Move in a zig-zag way over all nodes of the $(2t)$-gon.
  - Move first to the direct right neighbour,
  - and then to the direct left neighbour (i.e. two nodes back).
  - Continue is the same fashion.
  - Connect the two opposing end-nodes through $a_{2t+1}$
- We may identify for each edge a unique 2-factor.
Proof II

**Theorem**

The graph $K_{2t+1}$ is the sum of $t$ spanning cycles.

- Draw $2t$ nodes $a_1, a_2, \ldots, a_{2t}$, as a regular $(2t)$-gon.
- Draw $a_{2t+1}$ as the top of a pyramid above the nodes $a_1, a_2, \ldots, a_{2t}$.
- Choose one 2-factor:
  - connect two opposing nodes as follows:
  - Move in a zig-zag way over all nodes of the $(2t)$-gon.
  - Move first to the direct right neighbour,
  - and then to the direct left neighbour (i.e. two nodes back).
  - Continue is the same fashion.
  - Connect the two opposing end-nodes through $a_{2t+1}$
- We may identify for each edge a unique 2-factor.
Theorem

The graph $K_{2t+1}$ is the sum of $t$ spanning cycles.

- Draw $2t$ nodes $a_1, a_2, \cdots, a_{2t}$, as a regular $(2t)$-gon.
- Draw $a_{2t+1}$ as the top of a pyramid above the nodes $a_1, a_2, \cdots, a_{2t}$.
- Choose one 2-factor:
  - connect two opposing nodes as follows:
  - Move in a zig-zag way over all nodes of the $(2t)$-gon.
  - Move first to the direct right neighbour,
  - and then to the direct left neighbour (i.e. two nodes back).
  - Continue in the same fashion.
  - Connect the two opposing end-nodes through $a_{2t+1}$
- We may identify for each edge a unique 2-factor.
Theorem

The graph $K_{2t+1}$ is the sum of $t$ spanning cycles.

- Draw $2t$ nodes $a_1, a_2, \ldots, a_{2t}$, as a regular $(2t)$-gon.
- Draw $a_{2t+1}$ as the top of a pyramid above the nodes $a_1, a_2, \ldots, a_{2t}$.
- Choose one 2-factor:
  - connect two opposing nodes as follows:
  - Move in a zig-zag way over all nodes of the $(2t)$-gon.
  - Move first to the direct right neighbour,
  - and then to the direct left neighbour (i.e. two nodes back).
  - Continue is the same fashion.
  - Connect the two opposing end-nodes through $a_{2t+1}$
- We may identify for each edge a unique 2-factor.
Proof II

Theorem

The graph $K_{2t+1}$ is the sum of $t$ spanning cycles.

- Draw $2t$ nodes $a_1, a_2, \cdots, a_{2t}$, as a regular $(2t)$-gon.
- Draw $a_{2t+1}$ as the top of a pyramid above the nodes $a_1, a_2, \cdots, a_{2t}$.
- Choose one 2-factor:
  - connect two opposing nodes as follows:
  - Move in a zig-zag way over all nodes of the $(2t)$-gon.
  - Move first to the direct right neighbour,
  - and then to the direct left neighbour (i.e. two nodes back).
  - Continue in the same fashion.
  - Connect the two opposing end-nodes through $a_{2t+1}$.
- We may identify for each edge a unique 2-factor.
Proof II

**Theorem**

*The graph $K_{2t+1}$ is the sum of $t$ spanning cycles.*

- Draw $2t$ nodes $a_1, a_2, \cdots, a_{2t}$, as a regular $(2t)$-gon.
- Draw $a_{2t+1}$ as the top of a pyramid above the nodes $a_1, a_2, \cdots, a_{2t}$.
- Choose one 2-factor:
  - connect two opposing nodes as follows:
  - Move in a zig-zag way over all nodes of the $(2t)$-gon.
  - Move first to the direct right neighbour,
  - and then to the direct left neighbour (i.e. two nodes back).
  - Continue is the same fashion.
  - Connect the two opposing end-nodes through $a_{2t+1}$
- We may identify for each edge a unique 2-factor.
**Theorem**

The graph $K_{2t+1}$ is the sum of $t$ spanning cycles.

- Draw $2t$ nodes $a_1, a_2, \cdots, a_{2t}$, as a regular $(2t)$-gon.
- Draw $a_{2t+1}$ as the top of a pyramid above the nodes $a_1, a_2, \cdots, a_{2t}$.
- Choose one 2-factor:
  - connect two opposing nodes as follows:
    - Move in a zig-zag way over all nodes of the $(2t)$-gon.
    - Move first to the direct right neighbour,
    - and then to the direct left neighbour (i.e. two nodes back).
    - Continue in the same fashion.
    - Connect the two opposing end-nodes through $a_{2t+1}$
- We may identify for each edge a unique 2-factor.
**Definition**

Let $G$ be a graph. A spanning graph $H$ of $G$ is called $[k, k']$-factor, iff for all nodes $v$ of $H$ we have: $k \leq \deg(v) \leq k'$. The $k, k'$-factor is called perfect, iff each connectivity component is regular.

**Theorem (Tutte 1953)**

A graph $G = (V, E)$ contains a perfect $[1,2]$-factor, iff for each $S \subset V$ hold: $|S| \leq |\Gamma(S)|$.

**Proof ($\Rightarrow$)**

- Let $S$ be a perfect $[1,2]$-factor.
- $S_1 = \{x \in S \mid \deg_S(x) = 1\}$ and $S_2 = \{x \in S \mid \deg_S(x) = 2\}$.
- Thus we get $|S_1| = |\Gamma_H(S_1)|$ and $|S_2| \leq |\Gamma_H(S_2)|$.
- Because $\Gamma_H(S_2)$ and $\Gamma_H(S_1)$ are disjoint, we get:
- $|S| = |S_1| + |S_2| \leq |\Gamma_H(S_1)| + |\Gamma_H(S_2)| = |\Gamma_H(S)| \leq |\Gamma_G(S)|$. 
Basic Definitions
Connectivity of Graphs
Flows
Matchings
Factors of Graphs
Posets

Statements (1:48.2)

Definition
Let $G$ be a graph. A spanning graph $H$ of $G$ is called $[k, k']$-factor, iff for all nodes $v$ of $H$ we have: $k \leq \deg(v) \leq k'$. The $k, k'$-factor is called perfect, iff each connectivity component is regular.

Theorem (Tutte 1953)

A graph $G = (V, E)$ contains a perfect $[1,2]$-factor, iff for each $S \subset V$ hold: $|S| \leq |\Gamma(S)|$.

Proof ($\rightarrow$)

- Let $S$ be a perfect $[1,2]$-factor.
- $S_1 = \{x \in S \mid \deg_S(x) = 1\}$ and $S_2 = \{x \in S \mid \deg_S(x) = 2\}$.
- Thus we get $|S_1| = |\Gamma_H(S_1)|$ and $|S_2| \leq |\Gamma_H(S_2)|$.
- Because $\Gamma_H(S_2)$ and $\Gamma_H(S_1)$ are disjoint, we get:
- $|S| = |S_1| + |S_2| \leq |\Gamma_H(S_1)| + |\Gamma_H(S_2)| = |\Gamma_H(S)| \leq |\Gamma_G(S)|$. 
Definition

Let $G$ be a graph. A spanning graph $H$ of $G$ is called $[k, k']$-factor, iff for all nodes $v$ of $H$ we have: $k \leq \deg(v) \leq k'$. The $k, k'$-factor is called perfect, iff each connectivity component is regular.

Theorem (Tutte 1953)

A graph $G = (V, E)$ contains a perfect $[1,2]$-factor, iff for each $S \subset V$ hold: $|S| \leq |\Gamma(S)|$.

Proof ($\Rightarrow$)

- Let $S$ be a perfect $[1,2]$-factor.
- $S_1 = \{x \in S \mid \deg_S(x) = 1\}$ and $S_2 = \{x \in S \mid \deg_S(x) = 2\}$.
- Thus we get $|S_1| = |\Gamma_H(S_1)|$ and $|S_2| \leq |\Gamma_H(S_2)|$.
- Because $\Gamma_H(S_2)$ and $\Gamma_H(S_1)$ are disjoint, we get:
- $|S| = |S_1| + |S_2| \leq |\Gamma_H(S_1)| + |\Gamma_H(S_2)| = |\Gamma_H(S)| \leq |\Gamma_G(S)|$. 
Definition

Let $G$ be a graph. A spanning graph $H$ of $G$ is called $[k, k']$-factor, iff for all nodes $v$ of $H$ we have: $k \leq \deg(v) \leq k'$. The $k, k'$-factor is called perfect, iff each connectivity component is regular.

Theorem (Tutte 1953)

A graph $G = (V, E)$ contains a perfect $[1,2]$-factor, iff for each $S \subset V$ hold: $|S| \leq |\Gamma(S)|$.

Proof ($\Rightarrow$)

- Let $S$ be a perfect $[1,2]$-factor.
- $S_1 = \{x \in S \mid \deg_S(x) = 1\}$ and $S_2 = \{x \in S \mid \deg_S(x) = 2\}$.
- Thus we get $|S_1| = |\Gamma_H(S_1)|$ and $|S_2| \leq |\Gamma_H(S_2)|$.
- Because $\Gamma_H(S_2)$ and $\Gamma_H(S_1)$ are disjoint, we get:
- $|S| = |S_1| + |S_2| \leq |\Gamma_H(S_1)| + |\Gamma_H(S_2)| = |\Gamma_H(S)| \leq |\Gamma_G(S)|$. 
Definition

Let $G$ be a graph. A spanning graph $H$ of $G$ is called $[k, k']$-factor, iff for all nodes $v$ of $H$ we have: $k \leq \deg(v) \leq k'$. The $k, k'$-factor is called perfect, iff each connectivity component is regular.

Theorem (Tutte 1953)

A graph $G = (V, E)$ contains a perfect $[1,2]$-factor, iff for each $S \subset V$ hold: $|S| \leq |\Gamma(S)|$.

Proof ($\Rightarrow$)

- Let $S$ be a perfect $[1,2]$-factor.
- $S_1 = \{x \in S \mid \deg_S(x) = 1\}$ and $S_2 = \{x \in S \mid \deg_S(x) = 2\}$.
- Thus we get $|S_1| = |\Gamma_H(S_1)|$ and $|S_2| \leq |\Gamma_H(S_2)|$.
- Because $\Gamma_H(S_2)$ and $\Gamma_H(S_1)$ are disjoint, we get:
- $|S| = |S_1| + |S_2| \leq |\Gamma_H(S_1)| + |\Gamma_H(S_2)| = |\Gamma_H(S)| \leq |\Gamma_G(S)|$. 
Definition

Let $G$ be a graph. A spanning graph $H$ of $G$ is called $[k, k']$-factor, iff for all nodes $v$ of $H$ we have: $k \leq \deg(v) \leq k'$. The $k, k'$-factor is called perfect, iff each connectivity component is regular.

Theorem (Tutte 1953)

A graph $G = (V, E)$ contains a perfect $[1,2]$-factor, iff for each $S \subseteq V$ hold: $|S| \leq |\Gamma(S)|$.

Proof ($\Longrightarrow$)

- Let $S$ be a perfect $[1,2]$-factor.
- $S_1 = \{x \in S \mid \deg_S(x) = 1\}$ and $S_2 = \{x \in S \mid \deg_S(x) = 2\}$.
- Thus we get $|S_1| = |\Gamma_H(S_1)|$ and $|S_2| \leq |\Gamma_H(S_2)|$.
- Because $\Gamma_H(S_2)$ and $\Gamma_H(S_1)$ are disjoint, we get:
- $|S| = |S_1| + |S_2| \leq |\Gamma_H(S_1)| + |\Gamma_H(S_2)| = |\Gamma_H(S)| \leq |\Gamma_G(S)|$. 
Definition

Let $G$ be a graph. A spanning graph $H$ of $G$ is called $[k, k']$-factor, iff for all nodes $v$ of $H$ we have: $k \leq \deg(v) \leq k'$. The $k, k'$-factor is called perfect, iff each connectivity component is regular.

Theorem (Tutte 1953)

A graph $G = (V, E)$ contains a perfect $[1, 2]$-factor, iff for each $S \subset V$ hold: $|S| \leq |\Gamma(S)|$.

Proof ($\Rightarrow$)

- Let $S$ be a perfect $[1, 2]$-factor.
- $S_1 = \{x \in S \mid \deg_S(x) = 1\}$ and $S_2 = \{x \in S \mid \deg_S(x) = 2\}$.
- Thus we get $|S_1| = |\Gamma_H(S_1)|$ and $|S_2| \leq |\Gamma_H(S_2)|$.
- Because $\Gamma_H(S_2)$ and $\Gamma_H(S_1)$ are disjoint, we get:
  - $|S| = |S_1| + |S_2| \leq |\Gamma_H(S_1)| + |\Gamma_H(S_2)| = |\Gamma_H(S)| \leq |\Gamma_G(S)|$. 
Definition

Let $G$ be a graph. A spanning graph $H$ of $G$ is called a $[k, k']$-factor, iff for all nodes $v$ of $H$ we have: $k \leq \deg(v) \leq k'$. The $k, k'$-factor is called perfect, iff each connectivity component is regular.

Theorem (Tutte 1953)

A graph $G = (V, E)$ contains a perfect $[1,2]$-factor, iff for each $S \subset V$ hold: $|S| \leq |\Gamma(S)|$.

Proof ($\Rightarrow$)

- Let $S$ be a perfect $[1,2]$-factor.
- $S_1 = \{x \in S \mid \deg_S(x) = 1\}$ and $S_2 = \{x \in S \mid \deg_S(x) = 2\}$.
- Thus we get $|S_1| = |\Gamma_H(S_1)|$ and $|S_2| \leq |\Gamma_H(S_2)|$.
- Because $\Gamma_H(S_2)$ and $\Gamma_H(S_1)$ are disjoint, we get:
- $|S| = |S_1| + |S_2| \leq |\Gamma_H(S_1)| + |\Gamma_H(S_2)| = |\Gamma_H(S)| \leq |\Gamma_G(S)|$. 

Proof (Part 2)

Theorem (Tutte 1953)

A graph $G = (V, E)$ contains a perfect $[1,2]$-factor, iff for each $S \subset V$ hold: $|S| \leq |\Gamma(S)|$.

Proof ($\Longleftarrow$):

- Let $V = \{x_1, x_2, \ldots, x_n\}$, and define: $V_1 = \{x'_1, x'_2, \ldots, x'_n\}$ and $V_2 = \{x''_1, x''_2, \ldots, x''_n\}$.
- $G' = (V_1, V_2, \{(x'_i, x''_j) \mid (x_i, x_j) \in E\})$ is a bipartite graph.
- Let $S' = \{x'_i \mid x_i \in S\}$.
- Then we get: $\Gamma(S') = \{x''_i \mid x_i \in \Gamma(S)\}$
- And: $|S'| = |S| \leq |\Gamma(S)| = |\Gamma(S')|$
- Thus $G'$ contains a 1-factor $M$ (matching).
- Let $H = \{(x_i, x_j) \mid (x'_i, x''_j) \in M\}$.
- Then is the graph $H$ a $[1,2]$-factor.
Proof (Part 2)

**Theorem (Tutte 1953)**

A graph $G = (V, E)$ contains a perfect $[1,2]$-factor, iff for each $S \subset V$ hold: $|S| \leq |\Gamma(S)|$.

Proof ($\iff$):

- Let $V = \{x_1, x_2, \cdots, x_n\}$, and define: $V_1 = \{x'_1, x'_2, \cdots, x'_n\}$ and $V_2 = \{x''_1, x''_2, \cdots, x''_n\}$.

- $G' = (V_1, V_2, \{(x'_i, x''_j) \mid (x_i, x_j) \in E\})$ is a bipartite graph.

- Let $S' = \{x'_i \mid x_i \in S\}$.

- Then we get: $\Gamma(S') = \{x''_i \mid x_i \in \Gamma(S)\}$

- And: $|S'| = |S| \leq |\Gamma(S)| = |\Gamma(S')|$

- Thus $G'$ contains a 1-factor $M$ (matching).

- Let $H = \{(x_i, x_j) \mid (x'_i, x''_j) \in M\}$.

- Then is the graph $H$ a $[1,2]$-factor.
Theorem (Tutte 1953)

A graph $G = (V, E)$ contains a perfect $[1,2]$-factor, iff for each $S \subset V$ hold: $|S| \leq |\Gamma(S)|$.

Proof ($\iff$):

- Let $V = \{x_1, x_2, \ldots, x_n\}$, and define: $V_1 = \{x'_1, x'_2, \ldots, x'_n\}$ and $V_2 = \{x''_1, x''_2, \ldots, x''_n\}$.
- $G' = (V_1, V_2, \{(x'_i, x''_j) \mid (x_i, x_j) \in E\})$ is a bipartite graph.
- Let $S' = \{x'_i \mid x_i \in S\}$.
- Then we get: $\Gamma(S') = \{x''_i \mid x_i \in \Gamma(S)\}$
- And: $|S'| = |S| \leq |\Gamma(S)| = |\Gamma(S')|$
- Thus $G'$ contains a 1-factor $M$ (matching).
- Let $H = \{(x_i, x_j) \mid (x'_i, x''_j) \in M\}$.
- Then is the graph $H$ a $[1,2]$-factor.
Theorem (Tutte 1953)

A graph \( G = (V, E) \) contains a perfect \([1,2]\)-factor, iff for each \( S \subset V \) hold:
\[
|S| \leq |\Gamma(S)|.
\]

Proof (\( \leftarrow \rightarrow \)):

- Let \( V = \{x_1, x_2, \ldots, x_n\} \), and define: \( V_1 = \{x'_1, x'_2, \ldots, x'_n\} \) and
  \( V_2 = \{x''_1, x''_2, \ldots, x''_n\} \).
- \( G' = (V_1, V_2, \{(x'_i, x''_j) \mid (x_i, x_j) \in E\}) \) is a bipartite graph.
- Let \( S' = \{x'_i \mid x_i \in S\} \).
- Then we get: \( \Gamma(S') = \{x''_i \mid x_i \in \Gamma(S)\} \)
- And: \( |S'| = |S| \leq |\Gamma(S)| = |\Gamma(S')| \)
- Thus \( G' \) contains a 1-factor \( M \) (matching).
- Let \( H = \{(x_i, x_j) \mid (x'_i, x''_j) \in M\} \).
- Then is the graph \( H \) a \([1,2]\)-factor.
Proof (Part 2)

Theorem (Tutte 1953)

A graph $G = (V, E)$ contains a perfect $[1,2]$-factor, iff for each $S \subseteq V$ hold: $|S| \leq |\Gamma(S)|$.

Proof ($\iff$):

- Let $V = \{x_1, x_2, \cdots, x_n\}$, and define: $V_1 = \{x'_1, x'_2, \cdots, x'_n\}$ and $V_2 = \{x''_1, x''_2, \cdots, x''_n\}$.

- $G' = (V_1, V_2, \{ (x'_i, x''_j) \mid (x_i, x_j) \in E \} )$ is a bipartite graph.

- Let $S' = \{x'_i \mid x_i \in S\}$.

- Then we get: $\Gamma(S') = \{x''_i \mid x_i \in \Gamma(S)\}$

- And: $|S'| = |S| \leq |\Gamma(S)| = |\Gamma(S')|$

- Thus $G'$ contains a 1-factor $M$ (matching).

- Let $H = \{ (x_i, x_j) \mid (x'_i, x''_j) \in M \}$.

- Then is the graph $H$ a $[1,2]$-factor.
Proof (Part 2)

Theorem (Tutte 1953)

A graph \( G = (V, E) \) contains a perfect \([1,2]\)-factor, iff for each \( S \subset V \) hold:
\( \vert S \vert \leq \vert \Gamma(S) \vert \).

Proof (\( \Leftarrow \)):

- Let \( V = \{x_1, x_2, \cdots, x_n\} \), and define: \( V_1 = \{x_1', x_2', \cdots, x_n'\} \) and \( V_2 = \{x_1'', x_2'', \cdots, x_n''\} \).
- \( G' = (V_1, V_2, \{(x_i', x_j'') \mid (x_i, x_j) \in E\}) \) is a bipartite graph.
- Let \( S' = \{x_i' \mid x_i \in S\} \).
- Then we get: \( \Gamma(S') = \{x_i'' \mid x_i \in \Gamma(S)\} \)
- And: \( \vert S' \vert = \vert S \vert \leq \vert \Gamma(S) \vert = \vert \Gamma(S') \vert \)
- Thus \( G' \) contains a 1-factor \( M \) (matching).
- Let \( H = \{(x_i, x_j) \mid (x_i', x_j'') \in M\} \).
- Then is the graph \( H \) a \([1,2]\)-factor.
Proof (Part 2)

Theorem (Tutte 1953)

A graph $G = (V, E)$ contains a perfect $[1,2]$-factor, iff for each $S \subseteq V$ hold: $|S| \leq |\Gamma(S)|$.

Proof ($\iff$):

- Let $V = \{x_1, x_2, \ldots, x_n\}$, and define: $V_1 = \{x'_1, x'_2, \ldots, x'_n\}$ and $V_2 = \{x''_1, x''_2, \ldots, x''_n\}$.
- $G' = (V_1, V_2, \{(x'_i, x''_j) \mid (x_i, x_j) \in E\})$ is a bipartite graph.
- Let $S' = \{x'_i \mid x_i \in S\}$.
- Then we get: $\Gamma(S') = \{x''_i \mid x_i \in \Gamma(S)\}$
- And: $|S'| = |S| \leq |\Gamma(S)| = |\Gamma(S')|$
- Thus $G'$ contains a 1-factor $M$ (matching).
- Let $H = \{(x_i, x_j) \mid (x'_i, x''_j) \in M\}$.
- Then is the graph $H$ a $[1,2]$-factor.
Proof (Part 2)

**Theorem (Tutte 1953)**

A graph $G = (V, E)$ contains a perfect $[1,2]$-factor, iff for each $S \subseteq V$ hold: $|S| \leq |\Gamma(S)|$.

Proof ($\Longleftrightarrow$):

- Let $V = \{x_1, x_2, \ldots, x_n\}$, and define: $V_1 = \{x'_1, x'_2, \ldots, x'_n\}$ and $V_2 = \{x''_1, x''_2, \ldots, x''_n\}$.
- $G' = (V_1, V_2, \{(x'_i, x''_j) \mid (x_i, x_j) \in E\})$ is a bipartite graph.
- Let $S' = \{x'_i \mid x_i \in S\}$.
- Then we get: $\Gamma(S') = \{x''_i \mid x_i \in \Gamma(S)\}$
- And: $|S'| = |S| \leq |\Gamma(S)| = |\Gamma(S')|$
- Thus $G'$ contains a 1-factor $M$ (matching).
- Let $H = \{(x_i, x_j) \mid (x'_i, x''_j) \in M\}$.
- Then is the graph $H$ a $[1,2]$-factor.
Theorem (Tutte 1953)

A graph \( G = (V, E) \) contains a perfect \([1,2]\)-factor, iff for each \( S \subset V \) hold:
\[ |S| \leq |\Gamma(S)|. \]

Proof (\(\Longleftarrow\)):

- Let \( V = \{x_1, x_2, \ldots, x_n\} \), and define: \( V_1 = \{x_1', x_2', \ldots, x_n'\} \) and \( V_2 = \{x_1'', x_2'', \ldots, x_n''\} \).
- \( G' = (V_1, V_2, \{(x_i', x_j'') \mid (x_i, x_j) \in E\}) \) is a bipartite graph.
- Let \( S' = \{x_i' \mid x_i \in S\} \).
- Then we get: \( \Gamma(S') = \{x_i'' \mid x_i \in \Gamma(S)\} \)
- And: \( |S'| = |S| \leq |\Gamma(S)| = |\Gamma(S')| \)
- Thus \( G' \) contains a 1-factor \( M \) (matching).
- Let \( H = \{(x_i, x_j) \mid (x_i', x_j'') \in M\} \).
- Then is the graph \( H \) a \([1,2]\)-factor.
Proof (Part 3):

Proof ($\iff$):

- Let $V = \{x_1, x_2, \ldots, x_n\}$, and define: $V_1 = \{x'_1, x'_2, \ldots, x'_n\}$ and $V_2 = \{x''_1, x''_2, \ldots, x''_n\}$.
- $G' = (V_1, V_2, \{(x'_i, x''_j) \mid (x_i, x_j) \in E\})$ is a bipartite graph.
- Let $S' = \{ x'_i \mid x_i \in S \}$.
- Then we get: $\Gamma(S') = \{ x''_i \mid x_i \in \Gamma(S) \}$
- And: $|S'| = |S| \leq |\Gamma(S)| = |\Gamma(S')|$
- Thus $G'$ contains a 1-factor $M$ (matching).
- Let $H = \{(x_i, x_j) \mid (x'_i, x''_j) \in M\}$.
- Then is the graph $H$ a [1,2]-factor.
- Show now: If $\deg_H(x_i) = 1$ and $\{x_i, x_j\} \in H$, then does $\deg_H(x_j) = 1$ hold:
  - There exist $k, l$: $(x'_i, x''_k), (x'_i, x''_l) \in M$.
  - Then we get $k = l$ and $\deg_H(x_j) = 1$. 
Proof (Part 3):

Proof ($\iff$):

- Let $V = \{x_1, x_2, \ldots, x_n\}$, and define: $V_1 = \{x'_1, x'_2, \ldots, x'_n\}$ and $V_2 = \{x''_1, x''_2, \ldots, x''_n\}$.
- $G' = (V_1, V_2, \{(x'_i, x''_j) \mid (x_i, x_j) \in E\})$ is a bipartite graph.
- Let $S' = \{x'_i \mid x_i \in S\}$.
- Then we get: $\Gamma(S') = \{x''_i \mid x_i \in \Gamma(S)\}$
- And: $|S'| = |S| \leq |\Gamma(S)| = |\Gamma(S')|$
- Thus $G'$ contains a 1-factor $M$ (matching).
- Let $H = \{(x_i, x_j) \mid (x'_i, x''_j) \in M\}$.
- Then is the graph $H$ a [1,2]-factor.
- Show now: If $\deg_H(x_i) = 1$ and $\{x_i, x_j\} \in H$, then does $\deg_H(x_j) = 1$ hold:
  - There exist $k, l$: $(x'_i, x''_k), (x'_i, x''_l) \in M$.
  - Then we get $k = l$ and $\deg_H(x_j) = 1$. 
Proof (Part 3):

Proof (⇐):

- Let $V = \{x_1, x_2, \ldots, x_n\}$, and define: $V_1 = \{x_1', x_2', \ldots, x_n'\}$ and $V_2 = \{x_1'', x_2'', \ldots, x_n''\}$.
- $G' = (V_1, V_2, \{(x_i', x_j'') \mid (x_i, x_j) \in E\})$ is a bipartite graph.
- Let $S' = \{x_i' \mid x_i \in S\}$.
- Then we get: $\Gamma(S') = \{x_i'' \mid x_i \in \Gamma(S)\}$
- And: $|S'| = |S| \leq |\Gamma(S)| = |\Gamma(S')|$
- Thus $G'$ contains a 1-factor $M$ (matching).
- Let $H = \{(x_i, x_j) \mid (x_i', x_j'') \in M\}$.
- Then is the graph $H$ a [1,2]-factor.
- Show now: If $\deg_H(x_i) = 1$ and $\{x_i, x_j\} \in H$, then does $\deg_H(x_j) = 1$ hold:
  - There exist $k, l$: $(x_i', x_k''), (x_i', x_i'') \in M$.
  - Then we get $k = l$ and $\deg_H(x_j) = 1$.!
Definition

A connectivity component of a graph $G$ is called odd (reps. even), if it contains an odd (resp. even) number of nodes. Let $q(G)$ be the number of odd connectivity components of $G$.

Theorem (Tutte 1947)

A graph $G = (V, E)$ contains a 1-factor, iff for each $S \subset V$ we have:

$q(G - S) \leq |S|$.

Theorem (Petersen 1891)

Let $G$ be a 3-regular 2-edge connected graph. Then is $G$ the sum of a 1-factor and a 2-factor.

Theorem (Petersen 1891)

A graph $G = (V, E)$ is the sum of $k$ 2-Factors, iff $G$ is $2k$-regular.
A connectivity component of a graph $G$ is called odd (resp. even), if it contains an odd (resp. even) number of nodes. Let $q(G)$ be the number of odd connectivity components of $G$.

**Theorem (Tutte 1947)**

A graph $G = (V, E)$ contains a 1-factor, iff for each $S \subset V$ we have:

$$q(G - S) \leq |S|.$$

**Theorem (Petersen 1891)**

Let $G$ be a 3-regular 2-edge connected graph. Then is $G$ the sum of a 1-factor and a 2-factor.

**Theorem (Petersen 1891)**

A graph $G = (V, E)$ is the sum of $k$ 2-Factors, iff $G$ is $2k$-regular.
**Definition**

A connectivity component of a graph $G$ is called odd (reps. even), if it contains an odd (resp. even) number of nodes. Let $q(G)$ be the number of odd connectivity components of $G$.

**Theorem (Tutte 1947)**

A graph $G = (V, E)$ contains a 1-factor, iff for each $S \subset V$ we have: $q(G - S) \leq |S|$.

**Theorem (Petersen 1891)**

Let $G$ be a 3-regular 2-edge connected graph. Then is $G$ the sum of a 1-factor and a 2-factor.

**Theorem (Petersen 1891)**

A Graph $G = (V, E)$ is the sum of $k$ 2-Factors, iff $G$ is $2k$-regular.
**Definition**

A connectivity component of a graph $G$ is called odd (reps. even), if it contains an odd (resp. even) number of nodes. Let $q(G)$ be the number of odd connectivity components of $G$.

**Theorem (Tutte 1947)**

*A graph $G = (V, E)$ contains a 1-factor, iff for each $S \subseteq V$ we have: $q(G - S) \leq |S|$.*

**Theorem (Petersen 1891)**

*Let $G$ be a 3-regular 2-edge connected graph. Then is $G$ the sum of a 1-factor and a 2-factor.*

**Theorem (Petersen 1891)**

*A Graph $G = (V, E)$ is the sum of $k$ 2-Factors, iff $G$ is $2k$-regular.*
Proof I (Part 1)

**Theorem (Tutte 1947)**

A graph $G = (V, E)$ contains a 1-factor, iff for each $S \subseteq V$ we have:

$q(G - S) \leq |S|$. 

**Proof ($\implies$)**

- Let $S \subseteq V$ and $G$ has a 1-factor.
- Let $U_1, U_2, \cdots U_p$ be the odd components of $G - S$.
- From each $U_i$ must be an edge of the factor, which goes to $S$.
- Let $\{u_i, s_i\}$ be that edge.
- Then we get: $q(G - S) = p = |\{s_1, s_2, \cdots, s_p\}| \leq |S|$. 
Proof I (Part 1)

**Theorem (Tutte 1947)**

A graph $G = (V, E)$ contains a 1-factor, iff for each $S \subseteq V$ we have:
$q(G - S) \leq |S|$. 

Proof ($\implies$)

1. Let $S \subseteq V$ and $G$ has a 1-factor.
2. Let $U_1, U_2, \cdots U_p$ be the odd components of $G - S$.
3. From each $U_i$ must be an edge of the factor, which goes to $S$.
4. Let $\{u_i, s_i\}$ be that edge.
5. Then we get: $q(G - S) = p = |\{s_1, s_2, \cdots, s_p\}| \leq |S|$. 
Theorem (Tutte 1947)

A graph $G = (V, E)$ contains a 1-factor, iff for each $S \subset V$ we have:
$q(G - S) \leq |S|$. 

Proof ($\Rightarrow$)

- Let $S \subset V$ and $G$ has a 1-factor.
- Let $U_1, U_2, \cdots U_p$ be the odd components of $G - S$.
- From each $U_i$ must be an edge of the factor, which goes to $S$.
- Let $\{u_i, s_i\}$ be that edge.
- Then we get: $q(G - S) = p = |\{s_1, s_2, \cdots , s_p\}| \leq |S|$. 
Proof I (Part 1)

**Theorem (Tutte 1947)**

A graph \( G = (V, E) \) contains a 1-factor, iff for each \( S \subseteq V \) we have:

\[
q(G - S) \leq |S|.
\]

Proof (\( \implies \))

- Let \( S \subseteq V \) and \( G \) has a 1-factor.
- Let \( U_1, U_2, \cdots U_p \) be the odd components of \( G - S \).
- From each \( U_i \) must be an edge of the factor, which goes to \( S \).
- Let \( \{u_i, s_i\} \) be that edge.
- Then we get: \( q(G - S) = p = |\{s_1, s_2, \cdots, s_p\}| \leq |S| \).
Proof I (Part 1)

**Theorem (Tutte 1947)**

A graph $G = (V, E)$ contains a 1-factor, iff for each $S \subset V$ we have:
\[ q(G - S) \leq |S|. \]

Proof (\(\Rightarrow\))

- Let $S \subset V$ and $G$ has a 1-factor.
- Let $U_1, U_2, \cdots U_p$ be the odd components of $G - S$.
- From each $U_i$ must be an edge of the factor, which goes to $S$.
- Let $\{u_i, s_i\}$ be that edge.
- Then we get: $q(G - S) = p = |\{s_1, s_2, \cdots , s_p\}| \leq |S|$. 
Proof I (Part 1)

Theorem (Tutte 1947)

A graph $G = (V, E)$ contains a 1-factor, iff for each $S \subseteq V$ we have:
$q(G - S) \leq |S|$. 

Proof ($\implies$)

- Let $S \subseteq V$ and $G$ has a 1-factor.
- Let $U_1, U_2, \cdots U_p$ be the odd components of $G - S$.
- From each $U_i$ must be an edge of the factor, which goes to $S$.
- Let $\{u_i, s_i\}$ be that edge.
- Then we get: $q(G - S) = p = |\{s_1, s_2, \cdots, s_p\}| \leq |S|$.
Theorem (Tutte 1947)

A graph $G = (V, E)$ contains a 1-factor, iff for each $S \subseteq V$ we have:
$q(G - S) \leq |S|$. 

Proof ($\Leftarrow$) by induction over $n = |V|$: 

- Note: For all odd $n$ holds the statement.
- Note also for this: $S = \emptyset$.
- Start of induction $n = 2$:
- Because of $S = \emptyset$ is there an edge.
- Thus we have the start of the induction.
Proof I (Part 2)

Theorem (Tutte 1947)

**A graph** $G = (V, E)$ **contains a 1-factor, iff for each** $S \subset V$ **we have:**

$q(G - S) \leq |S|.$

Proof ($\iff$) by induction over $n = |V|:

- Note: For all odd $n$ holds the statement.
- **Note also for this:** $S = \emptyset$.
- Start of induction $n = 2$:
  - Because of $S = \emptyset$ is there an edge.
  - Thus we have the start of the induction.
Proof (Part 2)

Theorem (Tutte 1947)

A graph $G = (V, E)$ contains a 1-factor, iff for each $S \subseteq V$ we have:
$q(G - S) \leq |S|$.  

Proof ($\iff$) by induction over $n = |V|$:  

- Note: For all odd $n$ holds the statement.  
- Note also for this: $S = \emptyset$.  
- Start of induction $n = 2$:  
  - Because of $S = \emptyset$ is there an edge.  
  - Thus we have the start of the induction.
Proof (Part 2)

### Theorem (Tutte 1947)

A graph $G = (V, E)$ contains a 1-factor, iff for each $S \subseteq V$ we have:

$$q(G - S) \leq |S|.$$ 

Proof ($\iff$) by induction over $n = |V|$:

- Note: For all odd $n$ holds the statement.
- Note also for this: $S = \emptyset$.
- Start of induction $n = 2$:
  - Because of $S = \emptyset$ is there an edge.
  - Thus we have the start of the induction.
Proof I (Part 2)

Theorem (Tutte 1947)

A graph $G = (V, E)$ contains a 1-factor, iff for each $S \subseteq V$ we have:
$$q(G - S) \leq |S|.$$ 

Proof ($\Leftarrow$) by induction over $n = |V|$:

- Note: For all odd $n$ holds the statement.
- Note also for this: $S = \emptyset$.
- Start of induction $n = 2$:
  - Because of $S = \emptyset$ is there an edge.
  - Thus we have the start of the induction.
Proof I (Part 2)

Theorem (Tutte 1947)

A graph $G = (V, E)$ contains a 1-factor, iff for each $S \subset V$ we have:
$q(G - S) \leq |S|$.  

Proof ($\Leftarrow$) by induction over $n = |V|$:  

- Note: For all odd $n$ holds the statement.
- Note also for this: $S = \emptyset$.
- Start of induction $n = 2$:  
- Because of $S = \emptyset$ is there an edge.
- Thus we have the start of the induction.
Theorem (Tutte 1947)

A graph \( G = (V, E) \) contains a 1-factor, iff for each \( S \subset V \) we have:
\[ q(G - S) \leq |S|. \]

Proof (\( \iff \)) step of the induction \( n \geq 4 \):

- Choose \( S \) maximal with \( q(G - S) = |S| \)
- We show now that \( G - S \) contains no even components.
- Let \( U_1, U_2, \ldots, U_p \) be the odd components of \( G - S \).
- We show now, that for \( x_i \in V(U_i) \) the graph \( U_i - \{x_i\} \) has a 1-factor.
- After this we will find a 1-factor in \( G \).
Proof I (Part 3)

Theorem (Tutte 1947)

A graph $G = (V, E)$ contains a 1-factor, iff for each $S \subset V$ we have:
$q(G - S) \leq |S|$. 

Proof ($\Leftarrow$) step of the induction $n \geq 4$:

- Choose $S$ maximal with $q(G - S) = |S|$ 
- We show now that $G - S$ contains no even components. 
- Let $U_1, U_2, \cdots U_p$ be the odd components of $G - S$. 
- We show now, that for $x_i \in V(U_i)$ the graph $U_i - \{x_i\}$ has a 1-factor. 
- After this we will find a 1-factor in $G$. 
Proof I (Part 3)

**Theorem (Tutte 1947)**

A graph $G = (V, E)$ contains a 1-factor, iff for each $S \subset V$ we have:

$q(G - S) \leq |S|$.

Proof (⇐) step of the induction $n \geq 4$:

- Choose $S$ maximal with $q(G - S) = |S|$.
- We show now that $G - S$ contains no even components.
- Let $U_1, U_2, \cdots U_p$ be the odd components of $G - S$.
- We show now, that for $x_i \in V(U_i)$ the graph $U_i - \{x_i\}$ has a 1-factor.
- After this we will find a 1-factor in $G$. 
Proof I (Part 3)

Theorem (Tutte 1947)

A graph $G = (V, E)$ contains a 1-factor, iff for each $S \subseteq V$ we have:
$q(G - S) \leq |S|$. 

Proof ($\Leftarrow$) step of the induction $n \geq 4$:

- Choose $S$ maximal with $q(G - S) = |S|$
- We show now that $G - S$ contains no even components.
- Let $U_1, U_2, \cdots U_p$ be the odd components of $G - S$. 
- We show now, that for $x_i \in V(U_i)$ the graph $U_i - \{x_i\}$ has a 1-factor.
- After this we will find a 1-factor in $G$. 
Theorem (Tutte 1947)

A graph $G = (V, E)$ contains a 1-factor, iff for each $S \subseteq V$ we have: $q(G - S) \leq |S|$.

Proof ($\Leftarrow$) step of the induction $n \geq 4$:

- Choose $S$ maximal with $q(G - S) = |S|
- We show now that $G - S$ contains no even components.
- Let $U_1, U_2, \cdots U_p$ be the odd components of $G - S$.
- We show now, that for $x_i \in V(U_i)$ the graph $U_i - \{x_i\}$ has a 1-factor.
- After this we will find a 1-factor in $G$. 
Proof I (Part 3)

**Theorem (Tutte 1947)**

A graph $G = (V, E)$ contains a 1-factor, iff for each $S \subseteq V$ we have:

$$q(G - S) \leq |S|.$$ 

Proof ($\Leftarrow$) step of the induction $n \geq 4$:

- Choose $S$ maximal with $q(G - S) = |S|$
- We show now that $G - S$ contains no even components.
- Let $U_1, U_2, \ldots, U_p$ be the odd components of $G - S$.
- We show now, that for $x_i \in V(U_i)$ the graph $U_i - \{x_i\}$ has a 1-factor.
- After this we will find a 1-factor in $G$. 
Theorem (Tutte 1947)

A graph $G = (V, E)$ contains a 1-factor, iff for each $S \subset V$ we have: $q(G - S) \leq |S|$.

Show: $G - S$ contains no even components:

- Assume there is a even component $V'$ and $a \in V'$, then we get:
  - $|S| + 1 = 1 + q(G - S) \leq q(G - (S \cup \{a\})) \leq |S \cup \{a\}| = |S| + 1$
- This is a contradiction to the maximality of $S$. 
Proof I (Part 3a)

**Theorem (Tutte 1947)**

A graph $G = (V, E)$ contains a 1-factor, iff for each $S \subset V$ we have: $q(G - S) \leq |S|$.

Show: $G - S$ contains no even components:

1. Assume there is an even component $V'$ and $a \in V'$, then we get:

   $$|S| + 1 = 1 + q(G - S) \leq q(G - (S \cup \{a\})) \leq |S \cup \{a\}| = |S| + 1$$

2. This is a contradiction to the maximality of $S$. 
Proof I (Part 3a)

Theorem (Tutte 1947)

A graph $G = (V, E)$ contains a 1-factor, iff for each $S \subset V$ we have:

$q(G - S) \leq |S|$. 

Show: $G - S$ contains no even components:

- Assume there is an even component $V'$ and $a \in V'$, then we get:
  
  $|S| + 1 = 1 + q(G - S) \leq q(G - (S \cup \{a\})) \leq |S \cup \{a\}| = |S| + 1$

- This is a contradiction to the maximality of $S$. 

Proof I (Part 3a)

Theorem (Tutte 1947)

A graph \( G = (V, E) \) contains a 1-factor, iff for each \( S \subseteq V \) we have:
\[ q(G - S) \leq |S|. \]

Show: \( G - S \) contains no even components:

- Assume there is a even component \( V' \) and \( a \in V' \), then we get:
  \[ |S| + 1 = 1 + q(G - S) \leq q(G - (S \cup \{a\})) \leq |S \cup \{a\}| = |S| + 1 \]
- This is a contradiction to the maximality of \( S \).
Proof I (Part 3b)

Show: For $x_i \in V(U_i)$ has the graph $U_i - \{x_i\}$ a 1-factor.

- Assume, $H = U_i - \{x_i\}$ has no 1-factor.
- There exists $S' \subset V(H)$ with $q(H - S') > |S'|$.

Intermediate Step:

- $|V(H)|$ is even and $q(H - S') - |S'|$ is also even.
- If $|S'|$ is odd, then is also $|V(H) - S'|$ and $q(H - S')$ odd.
- If $|S'|$ is even, then is also $|V(H) - S'|$ and $q(H - S')$ even.

Then we have: $q(H - S') \geq |S'| + 2$.

- $|S| + |S'| + 1 = |S \cup S' \cup \{x_i\}| \geq q(G - (S \cup S' \cup \{x_i\}))$
- $q(G - (S \cup S' \cup \{x_i\})) = q(G - S) - 1 + q(H - S')$
- $q(G - S) - 1 + q(H - S') \geq |S| - 1 + |S'| + 2 = |S| + |S'| + 1$

This is a contradiction to the maximality of $S$. 
Proof I (Part 3b)

Show: For \( x_i \in V(U_i) \) has the graph \( U_i - \{x_i\} \) a 1-factor.

- Assume, \( H = U_i - \{x_i\} \) has no 1-factor.
- There exists \( S' \subset V(H) \) with \( q(H - S') > |S'| \).

Intermediate Step:

- \( |V(H)| \) is even and \( q(H - S') - |S'| \) is also even.
- If \( |S'| \) is odd, then is also \( |V(H) - S'| \) and \( q(H - S') \) odd.
- If \( |S'| \) is even, then is also \( |V(H) - S'| \) and \( q(H - S') \) even.

Then we have: \( q(H - S') \geq |S'| + 2 \).

\[
|S| + |S'| + 1 = |S \cup S' \cup \{x_i\}| \geq q(G - (S \cup S' \cup \{x_i\}))
\]

\[
q(G - (S \cup S' \cup \{x_i\})) = q(G - S) - 1 + q(H - S')
\]

\[
q(G - S) - 1 + q(H - S') \geq |S| - 1 + |S'| + 2 = |S| + |S'| + 1
\]

This is a contradiction to the maximality of \( S \).
Proof I (Part 3b)

Show: For $x_i \in V(U_i)$ has the graph $U_i - \{x_i\}$ a 1-factor.

- Assume, $H = U_i - \{x_i\}$ has no 1-factor.
- There exists $S' \subset V(H)$ with $q(H - S') > |S'|$.

**Intermediate Step:**

- $|V(H)|$ is even and $q(H - S') - |S'|$ is also even.
- If $|S'|$ is odd, then is also $|V(H) - S'|$ and $q(H - S')$ odd.
- If $|S'|$ is even, then is also $|V(H) - S'|$ and $q(H - S')$ even.

Then we have: $q(H - S') \geq |S'| + 2$.

- $|S| + |S'| + 1 = |S \cup S' \cup \{x_i\}| \geq q(G - (S \cup S' \cup \{x_i\}))$
- $q(G - (S \cup S' \cup \{x_i\})) = q(G - S) - 1 + q(H - S')$
- $q(G - S) - 1 + q(H - S') \geq |S| - 1 + |S'| + 2 = |S| + |S'| + 1$
- This is a contradiction to the maximality of $S$. 
Proof I (Part 3b)

Show: For $x_i \in V(U_i)$ has the graph $U_i - \{x_i\}$ a 1-factor.

- Assume, $H = U_i - \{x_i\}$ has no 1-factor.
- There exists $S' \subset V(H)$ with $q(H - S') > |S'|$.

Intermediate Step:

- $|V(H)|$ is even and $q(H - S') - |S'|$ is also even.
  - If $|S'|$ is odd, then is also $|V(H) - S'|$ and $q(H - S')$ odd.
  - If $|S'|$ is even, then is also $|V(H) - S'|$ and $q(H - S')$ even.

Then we have: $q(H - S') \geq |S'| + 2$.

- $|S| + |S'| + 1 = |S \cup S' \cup \{x_i\}| \geq q(G - (S \cup S' \cup \{x_i\}))$
- $q(G - (S \cup S' \cup \{x_i\})) = q(G - S) - 1 + q(H - S')$
- $q(G - S) - 1 + q(H - S') \geq |S| - 1 + |S'| + 2 = |S| + |S'| + 1$

This is a contradiction to the maximality of $S$. 
**Proof I (Part 3b)**

Show: For $x_i \in V(U_i)$ has the graph $U_i - \{x_i\}$ a 1-factor.

- Assume, $H = U_i - \{x_i\}$ has no 1-factor.
- There exists $S' \subset V(H)$ with $q(H - S') > |S'|$.

**Intermediate Step:**

- $|V(H)|$ is even and $q(H - S') - |S'|$ is also even.
- If $|S'|$ is odd, then is also $|V(H) - S'|$ and $q(H - S')$ odd.
- If $|S'|$ is even, then is also $|V(H) - S'|$ and $q(H - S')$ even.

Then we have: $q(H - S') \geq |S'| + 2$.
- $|S| + |S'| + 1 = |S \cup S' \cup \{x_i\}| \geq q(G - (S \cup S' \cup \{x_i\}))$
- $q(G - (S \cup S' \cup \{x_i\})) = q(G - S) - 1 + q(H - S')$
- $q(G - S) - 1 + q(H - S') \geq |S| - 1 + |S'| + 2 = |S| + |S'| + 1$
- This is a contradiction to the maximality of $S$. 
Proof I (Part 3b)

Show: For \( x_i \in V(U_i) \) has the graph \( U_i - \{x_i\} \) a 1-factor.

- Assume, \( H = U_i - \{x_i\} \) has no 1-factor.
- There exists \( S' \subset V(H) \) with \( q(H - S') > |S'| \).

Intermediate Step:

- \( |V(H)| \) is even and \( q(H - S') - |S'| \) is also even.
- If \( |S'| \) is odd, then is also \( |V(H) - S'| \) and \( q(H - S') \) odd.
- If \( |S'| \) is even, then is also \( |V(H) - S'| \) and \( q(H - S') \) even.

Then we have: \( q(H - S') \geq |S'| + 2 \).

\[
|S| + |S'| + 1 = |S \cup S' \cup \{x_i\}| \geq q(G - (S \cup S' \cup \{x_i\}))
\]

\[
q(G - (S \cup S' \cup \{x_i\})) = q(G - S) - 1 + q(H - S')
\]

\[
q(G - S) - 1 + q(H - S') \geq |S| - 1 + |S'| + 2 = |S| + |S'| + 1
\]

This is a contradiction to the maximality of \( S \).
Proof 1 (Part 3b)

Show: For \( x_i \in V(U_i) \) has the graph \( U_i - \{x_i\} \) a 1-factor.

- Assume, \( H = U_i - \{x_i\} \) has no 1-factor.
- There exists \( S' \subset V(H) \) with \( q(H - S') > |S'| \).

Intermediate Step:
- \( |V(H)| \) is even and \( q(H - S') - |S'| \) is also even.
- If \( |S'| \) is odd, then is also \( |V(H) - S'| \) and \( q(H - S') \) odd.
- If \( |S'| \) is even, then is also \( |V(H) - S'| \) and \( q(H - S') \) even.

Then we have: \( q(H - S') \geq |S'| + 2. \)

\[
|S| + |S'| + 1 = |S \cup S' \cup \{x_i\}| \geq q(G - (S \cup S' \cup \{x_i\}))
\]
\[
q(G - (S \cup S' \cup \{x_i\})) = q(G - S) - 1 + q(H - S')
\]
\[
q(G - S) - 1 + q(H - S') \geq |S| - 1 + |S'| + 2 = |S| + |S'| + 1
\]

This is a contradiction to the maximality of \( S \).
Proof I (Part 3b)

Show: For \( x_i \in V(U_i) \) has the graph \( U_i - \{x_i\} \) a 1-factor.

- Assume, \( H = U_i - \{x_i\} \) has no 1-factor.
- There exists \( S' \subset V(H) \) with \( q(H - S') > |S'| \).

Intermediate Step:
- \( |V(H)| \) is even and \( q(H - S') - |S'| \) is also even.
- If \( |S'| \) is odd, then is also \( |V(H) - S'| \) and \( q(H - S') \) odd.
- If \( |S'| \) is even, then is also \( |V(H) - S'| \) and \( q(H - S') \) even.

Then we have: \( q(H - S') \geq |S'| + 2 \).

- \( |S| + |S'| + 1 = |S \cup S' \cup \{x_i\}| \geq q(G - (S \cup S' \cup \{x_i\})) \)
- \( q(G - (S \cup S' \cup \{x_i\})) = q(G - S) - 1 + q(H - S') \)
- \( q(G - S) - 1 + q(H - S') \geq |S| - 1 + |S'| + 2 = |S| + |S'| + 1 \)
- This is a contradiction to the maximality of \( S \).
Proof I (Part 3b)

Show: For \( x_i \in V(U_i) \) has the graph \( U_i - \{x_i\} \) a 1-factor.

- Assume, \( H = U_i - \{x_i\} \) has no 1-factor.
- There exists \( S' \subset V(H) \) with \( q(H - S') > |S'| \).

Intermediate Step:

- \( |V(H)| \) is even and \( q(H - S') - |S'| \) is also even.
- If \( |S'| \) is odd, then is also \( |V(H) - S'| \) and \( q(H - S') \) odd.
- If \( |S'| \) is even, then is also \( |V(H) - S'| \) and \( q(H - S') \) even.

Then we have: \( q(H - S') \geq |S'| + 2 \).

\[ |S| + |S'| + 1 = |S \cup S' \cup \{x_i\}| \geq q(G - (S \cup S' \cup \{x_i\})) \]

\[ q(G - (S \cup S' \cup \{x_i\})) = q(G - S) - 1 + q(H - S') \]

\[ q(G - S) - 1 + q(H - S') \geq |S| - 1 + |S'| + 2 = |S| + |S'| + 1 \]

This is a contradiction to the maximality of \( S \).
Proof I (Part 3b)

Show: For \( x_i \in V(U_i) \) has the graph \( U_i - \{x_i\} \) a 1-factor.

- Assume, \( H = U_i - \{x_i\} \) has no 1-factor.
- There exists \( S' \subset V(H) \) with \( q(H - S') > |S'| \).

Intermediate Step:

- \( |V(H)| \) is even and \( q(H - S') - |S'| \) is also even.
- If \( |S'| \) is odd, then is also \( |V(H) - S'| \) and \( q(H - S') \) odd.
- If \( |S'| \) is even, then is also \( |V(H) - S'| \) and \( q(H - S') \) even.

Then we have: \( q(H - S') \geq |S'| + 2 \).

- \( |S| + |S'| + 1 = |S \cup S' \cup \{x_i\}| \geq q(G - (S \cup S' \cup \{x_i\})) \)
- \( q(G - (S \cup S' \cup \{x_i\})) = q(G - S) - 1 + q(H - S') \)

\( q(G - S) - 1 + q(H - S') \geq |S| - 1 + |S'| + 2 = |S| + |S'| + 1 \)

This is a contradiction to the maximality of \( S \).
Proof I (Part 3b)

Show: For $x_i \in V(U_i)$ has the graph $U_i - \{x_i\}$ a 1-factor.

- Assume, $H = U_i - \{x_i\}$ has no 1-factor.
- There exists $S' \subset V(H)$ with $q(H - S') > |S'|$.

Intermediate Step:

- $|V(H)|$ is even and $q(H - S') - |S'|$ is also even.
- If $|S'|$ is odd, then is also $|V(H) - S'|$ and $q(H - S')$ odd.
- If $|S'|$ is even, then is also $|V(H) - S'|$ and $q(H - S')$ even.

Then we have: $q(H - S') \geq |S'| + 2$.

- $|S| + |S'| + 1 = |S \cup S' \cup \{x_i\}| \geq q(G - (S \cup S' \cup \{x_i\}))$
- $q(G - (S \cup S' \cup \{x_i\})) = q(G - S) - 1 + q(H - S')$
- $q(G - S) - 1 + q(H - S') \geq |S| - 1 + |S'| + 2 = |S| + |S'| + 1$

This is a contradiction to the maximality of $S$. 
Proof I (Part 3b)

Show: For \( x_i \in V(U_i) \) has the graph \( U_i - \{x_i\} \) a 1-factor.

- Assume, \( H = U_i - \{x_i\} \) has no 1-factor.
- There exists \( S' \subset V(H) \) with \( q(H - S') > |S'| \).

Intermediate Step:
- \( |V(H)| \) is even and \( q(H - S') - |S'| \) is also even.
- If \( |S'| \) is odd, then is also \( |V(H) - S'| \) and \( q(H - S') \) odd.
- If \( |S'| \) is even, then is also \( |V(H) - S'| \) and \( q(H - S') \) even.

Then we have: \( q(H - S') \geq |S'| + 2 \).

\[
|S| + |S'| + 1 = |S \cup S' \cup \{x_i\}| \geq q(G - (S \cup S' \cup \{x_i\}))
\]

\[
q(G - (S \cup S' \cup \{x_i\})) = q(G - S) - 1 + q(H - S')
\]

\[
q(G - S) - 1 + q(H - S') \geq |S| - 1 + |S'| + 2 = |S| + |S'| + 1
\]

This is a contradiction to the maximality of \( S \).
Proof I (Part 3c)

Show: there is a 1-factor in $G$.

- Choose a matching $M$ with $|M| = p$ between $S$ and $U_1 \cup U_2 \cup \ldots \cup U_p$.
- Let: $U = \{U_1 \cup U_2 \cup \ldots \cup U_p\}$
- Let: $B = (U, S, \{\{U_i, s\} \mid \exists u_i \in V(U_i) : \{u_i, s\} \in E(G)\})$.
- Show that $B$ has a perfect matching.
- Let $X \subset U$ and $Y = \Gamma_B(X)$, then we have
  - $|X| \leq q(G - Y)$.
  - Put the above together: $|X| \leq q(G - Y) \leq |Y| = |\Gamma_B(X)|$.
- Thus $B$ has a perfect matching.
- Which is a 1-factor in $G$. 

Proof I (Part 3c)

Show: there is a 1-factor in \( G \).

- Choose a matching \( M \) with \(|M| = p\) between \( S \) and \( U_1 \cup U_2 \cup \ldots \cup U_p \).
- Let: \( U = \{U_1 \cup U_2 \cup \ldots \cup U_p\} \)
- Let: \( B = (U, S, \{\{U_i, s\} \mid \exists u_i \in V(U_i) : \{u_i, s\} \in E(G)\}) \).
- Show that \( B \) has a perfect matching.
- Let \( X \subset U \) and \( Y = \Gamma_B(X) \), then we have \(|X| \leq q(G - Y)\).
- Put the above together: \(|X| \leq q(G - Y) \leq |Y| = |\Gamma_B(X)|\).
- Thus \( B \) has a perfect matching.
- Which is a 1-factor in \( G \).
Proof I (Part 3c)

Show: there is a 1-factor in $G$.

1. Choose a matching $M$ with $|M| = p$ between $S$ and $U_1 \cup U_2 \cup \ldots \cup U_p$.
2. Let: $U = \{U_1 \cup U_2 \cup \ldots \cup U_p\}$
3. Let: $B = (U, S, \{\{U_i, s\} | \exists u_i \in V(U_i) : \{u_i, s\} \in E(G)\})$.
4. Show that $B$ has a perfect matching.
5. Let $X \subset U$ and $Y = \Gamma_B(X)$, then we have $|X| \leq q(G - Y)$.
6. Put the above together: $|X| \leq q(G - Y) \leq |Y| = |\Gamma_B(X)|$.
7. Thus $B$ has a perfect matching.
8. Which is a 1-factor in $G$. 
Proof I (Part 3c)

Show: there is a 1-factor in $G$.

- Choose a matching $M$ with $|M| = p$ between $S$ and $U_1 \cup U_2 \cup \ldots \cup U_p$.
- Let: $U = \{U_1 \cup U_2 \cup \ldots \cup U_p\}$
- Let: $B = (U, S, \{\{U_i, s\} \mid \exists u_i \in V(U_i) : \{u_i, s\} \in E(G)\})$.
- Show that $B$ has a perfect matching.

- Let $X \subset U$ and $Y = \Gamma_B(X)$, then we have $|X| \leq q(G - Y)$.
- Put the above together: $|X| \leq q(G - Y) \leq |Y| = |\Gamma_B(X)|$.
- Thus $B$ has a perfect matching.
- Which is a 1-factor in $G$. 
Proof I (Part 3c)

Show: there is a 1-factor in $G$.

- Choose a matching $M$ with $|M| = p$ between $S$ and $U_1 \cup U_2 \cup \ldots \cup U_p$.
- Let: $U = \{U_1 \cup U_2 \cup \ldots \cup U_p\}$
- Let: $B = (U, S, \{\{U_i, s\} | \exists u_i \in V(U_i) : \{u_i, s\} \in E(G)\})$.
- Show that $B$ has a perfect matching.
- Let $X \subset U$ and $Y = \Gamma_B(X)$, then we have
  - $|X| \leq q(G - Y)$.
  - Put the above together: $|X| \leq q(G - Y) \leq |Y| = |\Gamma_B(X)|$.
- Thus $B$ has a perfect matching.
- Which is a 1-factor in $G$. 
Proof I (Part 3c)

Show: there is a 1-factor in $G$.

- Choose a matching $M$ with $|M| = p$ between $S$ and $U_1 \cup U_2 \cup \ldots \cup U_p$.
- Let: $U = \{U_1 \cup U_2 \cup \ldots \cup U_p\}$
- Let: $B = (U, S, \{\{U_i, s\} | \exists u_i \in V(U_i) : \{u_i, s\} \in E(G)\})$.
- Show that $B$ has a perfect matching.
- Let $X \subset U$ and $Y = \Gamma_B(X)$, then we have
  - $|X| \leq q(G - Y)$.
  - Put the above together: $|X| \leq q(G - Y) \leq |Y| = |\Gamma_B(X)|$.
- Thus $B$ has a perfect matching.
- Which is a 1-factor in $G$. 
Proof 1 (Part 3c)

Show: there is a 1-factor in $G$.

- Choose a matching $M$ with $|M| = p$ between $S$ and $U_1 \cup U_2 \cup \ldots \cup U_p$.
- Let: $U = \{U_1 \cup U_2 \cup \ldots \cup U_p\}$
- Let: $B = (U, S, \{\{U_i, s\} | \exists u_i \in V(U_i) : \{u_i, s\} \in E(G)\})$.
- Show that $B$ has a perfect matching.
- Let $X \subset U$ and $Y = \Gamma_B(X)$, then we have
  - $|X| \leq q(G - Y)$.
  - Put the above together: $|X| \leq q(G - Y) \leq |Y| = |\Gamma_B(X)|$.
- Thus $B$ has a perfect matching.
- Which is a 1-factor in $G$. 
Proof I (Part 3c)

Show: there is a 1-factor in $G$.

- Choose a matching $M$ with $|M| = p$ between $S$ and $U_1 \cup U_2 \cup \ldots \cup U_p$.
- Let: $U = \{U_1 \cup U_2 \cup \ldots \cup U_p\}$
- Let: $B = (U, S, \{\{U_i, s\} | \exists u_i \in V(U_i) : \{u_i, s\} \in E(G)\})$.
- Show that $B$ has a perfect matching.
- Let $X \subset U$ and $Y = \Gamma_B(X)$, then we have
  - $|X| \leq q(G - Y)$.
  - Put the above together: $|X| \leq q(G - Y) \leq |Y| = |\Gamma_B(X)|$.
- Thus $B$ has a perfect matching.
  - Which is a 1-factor in $G$. 
Proof I (Part 3c)

Show: there is a 1-factor in $G$.

- Choose a matching $M$ with $|M| = p$ between $S$ and $U_1 \cup U_2 \cup \ldots \cup U_p$.
- Let: $U = \{U_1 \cup U_2 \cup \ldots \cup U_p\}$
- Let: $B = (U, S, \{\{U_i, s\} \mid \exists u_i \in V(U_i): \{u_i, s\} \in E(G)\})$.
- Show that $B$ has a perfect matching.
- Let $X \subset U$ and $Y = \Gamma_B(X)$, then we have
- $|X| \leq q(G - Y)$.
- Put the above together: $|X| \leq q(G - Y) \leq |Y| = |\Gamma_B(X)|$.
- Thus $B$ has a perfect matching.
- Which is a 1-factor in $G$. 
Proof I (Part 3c)

Show: there is a 1-factor in $G$.

- Choose a matching $M$ with $|M| = p$ between $S$ and $U_1 \cup U_2 \cup \ldots \cup U_p$.
- Let: $U = \{U_1 \cup U_2 \cup \ldots \cup U_p\}$
- Let: $B = (U, S, \{\{U_i, s\} | \exists u_i \in V(U_i) : \{u_i, s\} \in E(G)\})$.
- Show that $B$ has a perfect matching.
- Let $X \subset U$ and $Y = \Gamma_B(X)$, then we have
  - $|X| \leq q(G - Y)$.
  - Put the above together: $|X| \leq q(G - Y) \leq |Y| = |\Gamma_B(X)|$.
- Thus $B$ has a perfect matching.
- Which is a 1-factor in $G$. 
Proof II

Theorem (Petersen 1891)

A Graph $G = (V, E)$ is the sum of $k$ 2-Factors, iff $G$ is $2k$-regular.

$\implies$ trivial.

$\iff$ Induction and using Eulerian graph property.

- If $k = 1$ hold, then consists $G$ of disjoint cycles.
- $G$ has w.l.o.g. a Eulerian cycle.
- Direct the edges by the order of the Eulerian cycle (directed node set is $F$).
- Let $V = \{x_1, x_2, \cdots, x_n\}$ and define:
  
  $V_1 = \{x'_1, x'_2, \cdots, x'_n\}$ and $V_2 = \{x''_1, x''_2, \cdots, x''_n\}$.
- Thus $G' = (V_1, V_2, \{\{x'_i, x''_j\} \mid (x'_i, x''_j) \in F\})$ is a regular bipartite graph of degree $k$.
- This graph contains $k$ perfect matchings.
- These matchings give $k$ 2-factors in $G$. 
Proof II

Theorem (Petersen 1891)

A Graph \( G = (V, E) \) is the sum of \( k \) 2-Factors, iff \( G \) is \( 2k \)-regular.

\[ \implies \text{trivial.} \]

\[ \iff \text{Induction and using Eulerian graph property.} \]

- If \( k = 1 \) hold, then consists \( G \) of disjoint cycles.
- \( G \) has w.l.o.g. a Eulerian cycle.
- Direct the edges by the order of the Eulerian cycle (directed node set is \( F \)).
- Let \( V = \{x_1, x_2, \cdots, x_n\} \) and define:
  - \( V_1 = \{x'_1, x'_2, \cdots, x'_n\} \) and \( V_2 = \{x''_1, x''_2, \cdots, x''_n\} \).
- Thus \( G' = (V_1, V_2, \{\{x'_i, x''_j\} \mid (x'_i, x''_j) \in F\}) \) is a regular bipartite graph of degree \( k \).
- This graph contains \( k \) perfect matchings.
- These matchings give \( k \) 2-factors in \( G \).
Theorem (Petersen 1891)

A graph $G = (V, E)$ is the sum of $k$ 2-Factors, iff $G$ is $2k$-regular.

\[
\begin{align*}
\implies & \text{ trivial.} \\
\Leftarrow & \quad \text{Induction and using Eulerian graph property.}
\end{align*}
\]

- If $k = 1$ hold, then consists $G$ of disjoint cycles.
- $G$ has w.l.o.g. a Eulerian cycle.
- Direct the edges by the order of the Eulerian cycle (directed node set is $F$).
- Let $V = \{x_1, x_2, \ldots, x_n\}$ and define: $V_1 = \{x'_1, x'_2, \ldots, x'_n\}$ and $V_2 = \{x''_1, x''_2, \ldots, x''_n\}$.
- Thus $G' = (V_1, V_2, \{\{x'_i, x''_j\} \mid (x'_i, x''_j) \in F\})$ is a regular bipartite graph of degree $k$.
- This graph contains $k$ perfect matchings.
- These matchings give $k$ 2-factors in $G$. 
Proof II

Theorem (Petersen 1891)

A Graph \( G = (V, E) \) is the sum of \( k \) 2-Factors, iff \( G \) is 2\( k \)-regular.

\[ \implies \text{trivial.} \]

\[ \iff \text{Induction and using Eulerian graph property.} \]

- If \( k = 1 \) hold, then consists \( G \) of disjoint cycles.
- \( G \) has w.l.o.g. a Eulerian cycle.
- Direct the edges by the order of the Eulerian cycle (directed node set is \( F \)).
- Let \( V = \{x_1, x_2, \cdots, x_n\} \) and define:
  \[ V_1 = \{x'_1, x'_2, \cdots, x'_n\} \text{ and } V_2 = \{x''_1, x''_2, \cdots, x''_n\}. \]
  Thus \( G' = (V_1, V_2, \{\{x'_i, x''_j\} | (x'_i, x''_j) \in F\}) \) is a regular bipartite graph of degree \( k \).
- This graph contains \( k \) perfect matchings.
- These matchings give \( k \) 2-factors in \( G \).
Proof II

Theorem (Petersen 1891)

A Graph \( G = (V, E) \) is the sum of \( k \) 2-Factors, iff \( G \) is \( 2k \)-regular.

\[ \rightleftharpoons \text{trivial.} \]

\[ \leftarrow \text{Induction and using Eulerian graph property.} \]

- If \( k = 1 \) hold, then consists \( G \) of disjoint cycles.
- \( G \) has w.l.o.g. a Eulerian cycle.
- Direct the edges by the order of the Eulerian cycle (directed node set is \( F \)).
- Let \( V = \{x_1, x_2, \ldots, x_n\} \) and define:
  \( V_1 = \{x'_1, x'_2, \ldots, x'_n\} \) and \( V_2 = \{x''_1, x''_2, \ldots, x''_n\} \).
- Thus \( G' = (V_1, V_2, \{\{x'_i, x''_j\} \mid (x'_i, x''_j) \in F\}) \) is a regular bipartite graph of degree \( k \).
- This graph contains \( k \) perfect matchings.
- These matchings give \( k \) 2-factors in \( G \).
Theorem (Petersen 1891)

A Graph $G = (V, E)$ is the sum of $k$ 2-Factors, iff $G$ is 2$k$-regular.

$\implies$ trivial.

$\Leftarrow$ Induction and using Eulerian graph property.

- If $k = 1$ hold, then consists $G$ of disjoint cycles.
- $G$ has w.l.o.g. a Eulerian cycle.
- Direct the edges by the order of the Eulerian cycle (directed node set is $F$).
- Let $V = \{x_1, x_2, \ldots, x_n\}$ and define: $V_1 = \{x_1', x_2', \ldots, x_n'\}$ and $V_2 = \{x_1'', x_2'', \ldots, x_n''\}$.
- Thus $G' = (V_1, V_2, \{\{x_i', x_j''\} | (x_i', x_j'') \in F\}$ is a regular bipartite graph of degree $k$.
- This graph contains $k$ perfect matchings.
- These matchings give $k$ 2-factors in $G$. 
Theorem (Petersen 1891)

A Graph \( G = (V, E) \) is the sum of \( k \) 2-Factors, iff \( G \) is \( 2k \)-regular.

\[ \implies \text{trivial.} \]

\[ \iff \text{Induction and using Eulerian graph property.} \]

- If \( k = 1 \) hold, then consists \( G \) of disjoint cycles.
- \( G \) has w.l.o.g. a Eulerian cycle.
- Direct the edges by the order of the Eulerian cycle (directed node set is \( F \)).
- Let \( V = \{x_1, x_2, \cdots, x_n\} \) and define:
  \[ V_1 = \{x_1', x_2', \cdots, x_n'\} \text{ and } V_2 = \{x_1'', x_2'', \cdots, x_n''\}. \]
- Thus \( G' = (V_1, V_2, \{\{x_i', x_j''\} \mid (x_i', x_j'') \in F\}) \) is a regular bipartite graph of degree \( k \).
- This graph contains \( k \) perfect matchings.
- These matchings give \( k \) 2-factors in \( G \).
Proof II

Theorem (Petersen 1891)

A Graph $G = (V, E)$ is the sum of $k$ 2-Factors, iff $G$ is $2k$-regular.

$\implies$ trivial.

$\impliedby$ Induction and using Eulerian graph property.

- If $k = 1$ hold, then consists $G$ of disjoint cycles.
- $G$ has w.l.o.g. a Eulerian cycle.
- Direct the edges by the order of the Eulerian cycle (directed node set is $F$).
- Let $V = \{x_1, x_2, \cdots, x_n\}$ and define:
  \[ V_1 = \{x'_1, x'_2, \cdots, x'_n\} \text{ and } V_2 = \{x''_1, x''_2, \cdots, x''_n\}. \]
- Thus $G' = (V_1, V_2, \{\{x'_i, x''_j\} \mid (x'_i, x''_j) \in F\})$ is a regular bipartite graph of degree $k$.
- This graph contains $k$ perfect matchings.
- These matchings give $k$ 2-factors in $G$. 
Proof II

Theorem (Petersen 1891)

A Graph \( G = (V, E) \) is the sum of \( k \) 2-Factors, iff \( G \) is \( 2k \)-regular.

\[ \implies \text{trivial.} \]

\[ \iff \text{Induction and using Eulerian graph property.} \]

- If \( k = 1 \) hold, then consists \( G \) of disjoint cycles.
- \( G \) has w.l.o.g. a Eulerian cycle.
- Direct the edges by the order of the Eulerian cycle (directed node set is \( F \)).
- Let \( V = \{x_1, x_2, \ldots, x_n\} \) and define:
  \[ V_1 = \{x'_1, x'_2, \ldots, x'_n\} \text{ and } V_2 = \{x''_1, x''_2, \ldots, x''_n\}. \]
- Thus \( G' = (V_1, V_2, \{\{x'_i, x''_j\} \mid (x'_i, x''_j) \in F\}) \) is a regular bipartite graph of degree \( k \).
- This graph contains \( k \) perfect matchings.
- These matchings give \( k \) 2-factors in \( G \).
Proof II

Theorem (Petersen 1891)

A Graph $G = (V, E)$ is the sum of $k$ 2-Factors, iff $G$ is $2k$-regular.

$\implies$ trivial.

$\impliedby$ Induction and using Eulerian graph property.

- If $k = 1$ hold, then consists $G$ of disjoint cycles.
- $G$ has w.l.o.g. a Eulerian cycle.
- Direct the edges by the order of the Eulerian cycle (directed node set is $F$).
- Let $V = \{x_1, x_2, \ldots, x_n\}$ and define:
  $V_1 = \{x'_1, x'_2, \ldots, x'_n\}$ and $V_2 = \{x''_1, x''_2, \ldots, x''_n\}$.
- Thus $G' = (V_1, V_2, \{\{x'_i, x''_j\} \mid (x'_i, x''_j) \in F\})$ is a regular bipartite graph of degree $k$.
- This graph contains $k$ perfect matchings.
- These matchings give $k$ 2-factors in $G$. 
Theorem (Petersen 1891)

Let $G$ be a 3-regular 2-edge connected graph. Then is $G$ the sum of a 1-factor and a 2-factor.

- Let $A \subset V$.
- Let $U_1, U_2, \cdots, U_p$ be the odd components in $G - A$.
- For each component in $U_i$ exists at least 2 edges in $G$, who connect $U_i$ and $A$.
- Due to the 3-regularity are there at least 3 such edges.
- Thus there are at least $3 \cdot q(G - A)$ edges from $G - A$ to $A$.
- $3|A| = d_G(A) := \sum_{x \in A} d_G(x) \geq 3 \cdot q(G - A)$.
- $q(G - A) \leq |A|$.
- The proof is finished by using the Theorem of Tutte.
Proof III

Theorem (Petersen 1891)

Let $G$ be a 3-regular 2-edge connected graph. Then is $G$ the sum of a 1-factor and a 2-factor.

- Let $A \subseteq V$.
- Let $U_1, U_2, \cdots, U_p$ be the odd components in $G - A$.
- For each component in $U_i$ exists at least 2 edges in $G$, who connect $U_i$ and $A$.
- Due to the 3-regularity are there at least 3 such edges.
- Thus there are at least $3 \cdot q(G - A)$ edges from $G - A$ to $A$.
- $3|A| = d_G(A) := \sum_{x \in A} d_G(x) \geq 3 \cdot q(G - A)$.
- $q(G - A) \leq |A|$.
- The proof is finished by using the Theorem of Tutte.
Proof III

Theorem (Petersen 1891)

Let $G$ be a 3-regular 2-edge connected graph. Then is $G$ the sum of a 1-factor and a 2-factor.

- Let $A \subset V$.
- Let $U_1, U_2, \ldots, U_p$ be the odd components in $G - A$.
- For each component in $U_i$ exists at least 2 edges in $G$, who connect $U_i$ and $A$.
- Due to the 3-regularity are there at least 3 such edges.
- Thus there are at least $3 \cdot q(G - A)$ edges from $G - A$ to $A$.
- $3|A| = d_G(A) := \sum_{x \in A} d_G(x) \geq 3 \cdot q(G - A)$.
- $q(G - A) \leq |A|$.
- The proof is finished by using the Theorem of Tutte.
Proof III

Theorem (Petersen 1891)

Let $G$ be a 3-regular 2-edge connected graph. Then is $G$ the sum of a 1-factor and a 2-factor.

- Let $A \subset V$.
- Let $U_1, U_2, \cdots, U_p$ be the odd components in $G - A$.
- For each component in $U_i$ exists at least 2 edges in $G$, who connect $U_i$ and $A$.
- Due to the 3-regularity are there at least 3 such edges.
- Thus there are at least $3 \cdot q(G - A)$ edges from $G - A$ to $A$.
- $3|A| = d_G(A) := \sum_{x \in A} d_G(x) \geq 3 \cdot q(G - A)$.
- $q(G - A) \leq |A|$.
- The proof is finished by using the Theorem of Tutte.
Proof III

Theorem (Petersen 1891)

Let $G$ be a 3-regular 2-edge connected graph. Then is $G$ the sum of a 1-factor and a 2-factor.

- Let $A \subseteq V$.
- Let $U_1, U_2, \cdots, U_p$ be the odd components in $G - A$.
- For each component in $U_i$ exists at least 2 edges in $G$, who connect $U_i$ and $A$.
- Due to the 3-regularity are there at least 3 such edges.
- Thus there are at least $3 \cdot q(G - A)$ edges from $G - A$ to $A$.
- $3|A| = d_G(A) := \sum_{x \in A} d_G(x) \geq 3 \cdot q(G - A)$.
- $q(G - A) \leq |A|$.
- The proof is finished by using the Theorem of Tutte.
Theorem (Petersen 1891)

Let \( G \) be a 3-regular 2-edge connected graph. Then is \( G \) the sum of a 1-factor and a 2-factor.

- Let \( A \subset V \).
- Let \( U_1, U_2, \ldots, U_p \) be the odd components in \( G - A \).
- For each component in \( U_i \) exists at least 2 edges in \( G \), who connect \( U_i \) and \( A \).
- Due to the 3-regularity are there at least 3 such edges.
- Thus there are at least \( 3 \cdot q(G - A) \) edges from \( G - A \) to \( A \).
- \( 3|A| = d_G(A) := \sum_{x \in A} d_G(x) \geq 3 \cdot q(G - A) \).
- \( q(G - A) \leq |A| \).
- The proof is finished by using the Theorem of Tutte.
Proof III

Theorem (Petersen 1891)

Let $G$ be a 3-regular 2-edge connected graph. Then is $G$ the sum of a 1-factor and a 2-factor.

- Let $A \subseteq V$.
- Let $U_1, U_2, \cdots, U_p$ be the odd components in $G - A$.
- For each component in $U_i$ exists at least 2 edges in $G$, who connect $U_i$ and $A$.
- Due to the 3-regularity are there at least 3 such edges.
- Thus there are at least $3 \cdot q(G - A)$ edges from $G - A$ to $A$.
- $3|A| = d_G(A) := \sum_{x \in A} d_G(x) \geq 3 \cdot q(G - A)$.
- $q(G - A) \leq |A|$.
- The proof is finished by using the Theorem of Tutte.
Proof III

Theorem (Petersen 1891)

Let $G$ be a 3-regular 2-edge connected graph. Then $G$ is the sum of a 1-factor and a 2-factor.

- Let $A \subset V$.
- Let $U_1, U_2, \cdots, U_p$ be the odd components in $G - A$.
- For each component in $U_i$ exists at least 2 edges in $G$, who connect $U_i$ and $A$.
- Due to the 3-regularity are there at least 3 such edges.
- Thus there are at least $3 \cdot q(G - A)$ edges from $G - A$ to $A$.
- $3|A| = d_G(A) := \sum_{x \in A} d_G(x) \geq 3 \cdot q(G - A)$.
- $q(G - A) \leq |A|$.
- The proof is finished by using the Theorem of Tutte.
Proof III

Theorem (Petersen 1891)

Let $G$ be a 3-regular 2-edge connected graph. Then is $G$ the sum of a 1-factor and a 2-factor.

- Let $A \subset V$.
- Let $U_1, U_2, \cdots, U_p$ be the odd components in $G - A$.
- For each component in $U_i$ exists at least 2 edges in $G$, who connect $U_i$ and $A$.
- Due to the 3-regularity are there at least 3 such edges.
- Thus there are at least $3 \cdot q(G - A)$ edges from $G - A$ to $A$.
- $3|A| = d_G(A) := \sum_{x \in A} d_G(x) \geq 3 \cdot q(G - A)$.
- $q(G - A) \leq |A|$.
- The proof is finished by using the Theorem of Tutte.
Definition

Let $P$ be a finite set and $<$ be a transitive anti-reflexive relation. The pair $(P, <)$ is called a partly ordered set (poset). A subset $A \subset P$ is called an anti-chain, iff $x < y$ implies $\{x, y\} \notin A$. Furthermore, $C \subset P$ is called a chain, iff for all $x, y \in C$ holds either $x \leq y$ or $x > y$.

Theorem (Dilworth)

Let $P$ be a poset and $m$ is the cardinality of the largest anti-chain in $P$. Then is $P$ the union of $m$ chains.

Theorem (Sperner)

The cardinality of the maximal anti-chain in $Q^n$ is $\binom{n}{\lfloor n/2 \rfloor}$. 


Theorem (Leader 1995)

Let $A, B \subseteq Q^n$ with $|A| = \sum_{i=1}^{k} \binom{n}{i}$, $|B| = \sum_{i=1}^{l} \binom{n}{i}$ and $k \leq l < n/2$. Then we have:

- There are $\binom{n}{k}$ edges connecting $A$ with $Q^n \setminus A$;
- There are $\binom{n}{k}$ node disjoint paths from $A$ to $B$. 
Literature

1. Golumbic M.C. Algorithmic Graph Theory and Perfect Graphs
5. Bollobás B.: Extremal Graph Theory, 1976
Literature

1. Golumbic M.C. Algorithmic Graph Theory and Perfect Graphs
5. Bollobás B.: Extremal Graph Theory, 1976
Literature

1. Golumbic M.C. Algorithmic Graph Theory and Perfect Graphs
5. Bollobás B.: Extremal Graph Theory, 1976
Literature

1. Golumbic M.C. Algorithmic Graph Theory and Perfect Graphs
5. Bollobás B.: Extremal Graph Theory, 1976
Literature

1. Golumbic M.C. Algorithmic Graph Theory and Perfect Graphs
5. Bollobás B.: Extremal Graph Theory, 1976
Literature

1. Golumbic M.C. Algorithmic Graph Theory and Perfect Graphs
5. Bollobás B.: Extremal Graph Theory, 1976
### Literature

1. Golumbic M.C. *Algorithmic Graph Theory and Perfect Graphs*
Literature

1. Golumbic M.C. Algorithmic Graph Theory and Perfect Graphs
5. Bollobás B.: Extremal Graph Theory, 1976
1. Golumbic M.C. Algorithmic Graph Theory and Perfect Graphs
5. Bollobás B.: Extremal Graph Theory, 1976
Literature

1. Golumbic M.C. Algorithmic Graph Theory and Perfect Graphs
5. Bollobás B.: Extremal Graph Theory, 1976
Legend

- ■ : Not of relevance
- ■ : implicitly used basics
- ■ : idea of proof or algorithm
- ■ : structure of proof or algorithm
- ■ : Full knowledge