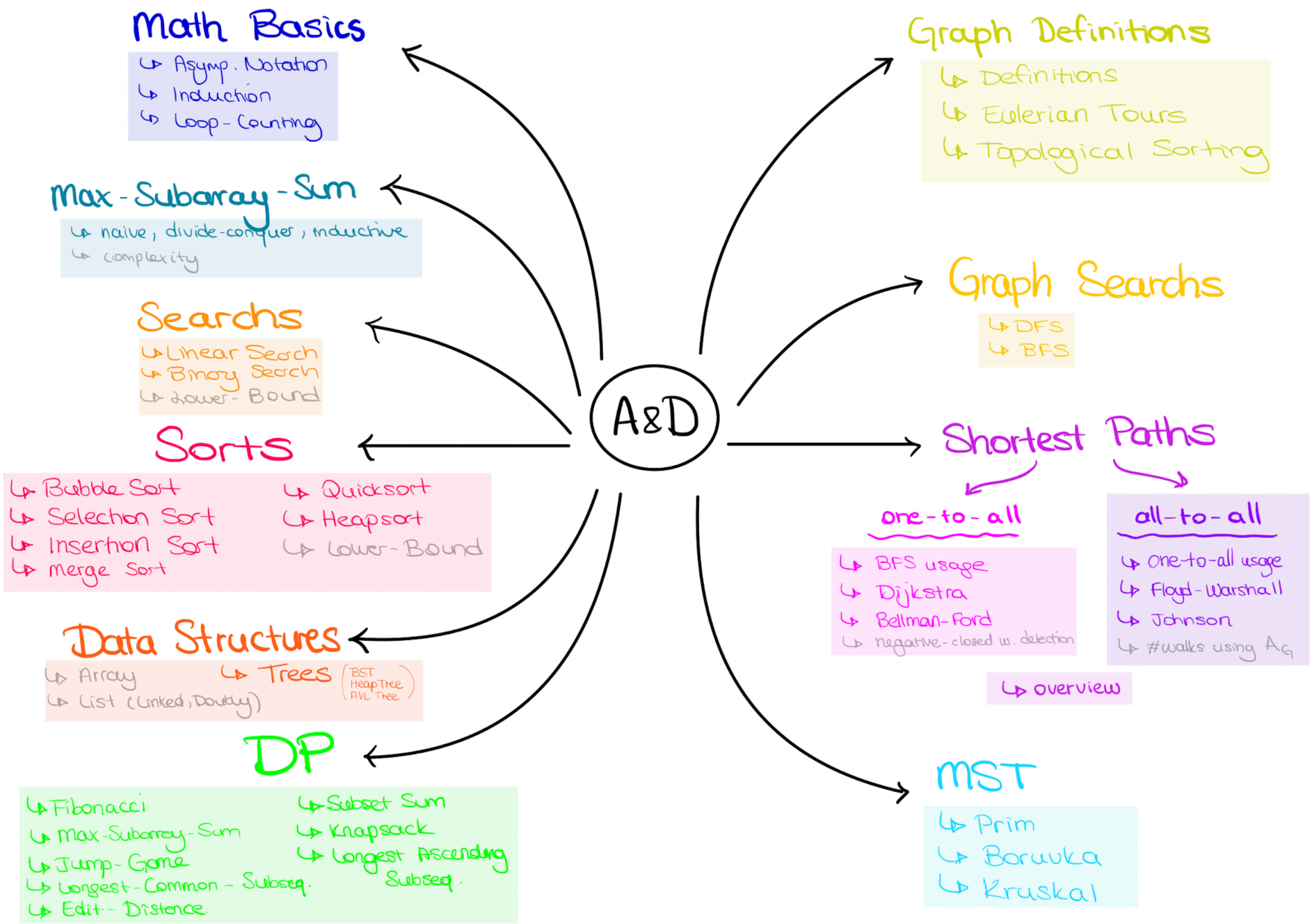
A&D

Exercise Session 8

Nil Ozer

A&D Overview


A&D
Math Basics
Asymp. Notation
Induction
Loop-Counting
Max-Subarray-Sum
naive, divide-conquer, inductive
complexity
Searchs
Linear Search
Binary Search
Lower-Bound
Sorts
Bubble Sort
Selection Sort
Insertion Sort
Merge Sort
Quicksort
Heapsort
Lower-Bound
Data Structures
Array
List (Linked, Doubly)
Trees
BST
Heap Tree
AVL Tree
DP
Fibonacci
Max-Subarray-Sum
Jump-Game
Longest-Common-Subseq.
Edit-Distance
Subset Sum
Knapsack
Longest Ascending Subseq.
Graph Definitions
Definitions
Eulerian Tours
Topological Sorting
Graph Searchs
DFS
BFS
Shortest Paths
one-to-all
BFS usage
Dijkstra
Bellman-Ford
negative-closed w. detection
all-to-all
one-to-all usage
Floyd-Warshall
Johnson
#walks using A_G
overview
MST
Prim
Boruvka
Kruskal

Outline

- Quiz
- Exercise Sheets

- Graph Definitions
- Short Proofs
- Exam Question

- Next week

Quiz

Mid Semester Feedback

Exercise Sheets

- Exercise Sheet 6 feedback left for next time
- Exercise Sheet 7 peergrading
  - 7.5 this week
  - Emails will be sent
- New groups for Exercise Sheet 7,8 and 9 !

Graph Definitions

Graph

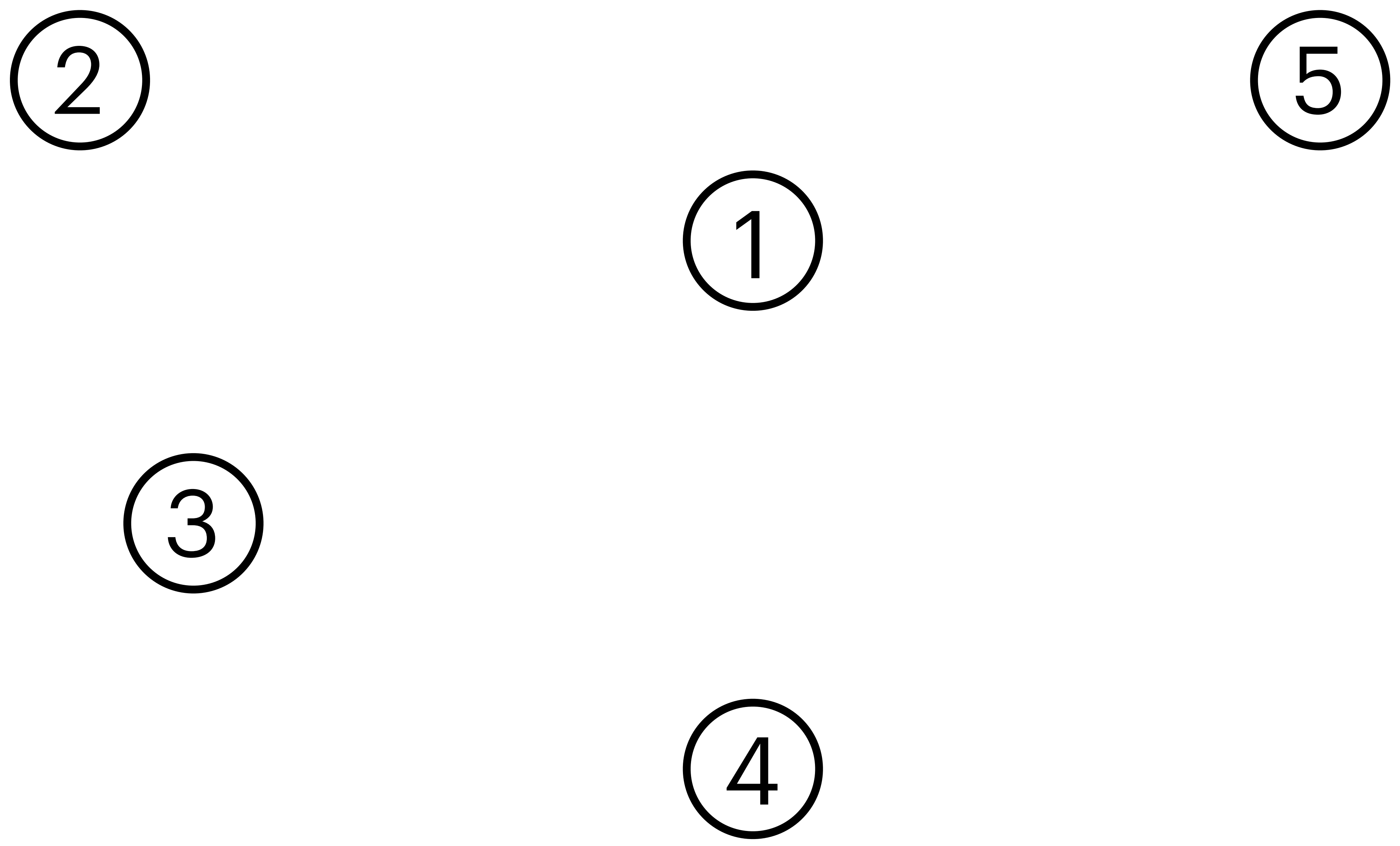
$$G = (V, E)$$

Graph

Node / Vertex

$G = (V, E)$

$V = \{v_1, \ldots, v_n\}, |V| = n$ the set of nodes

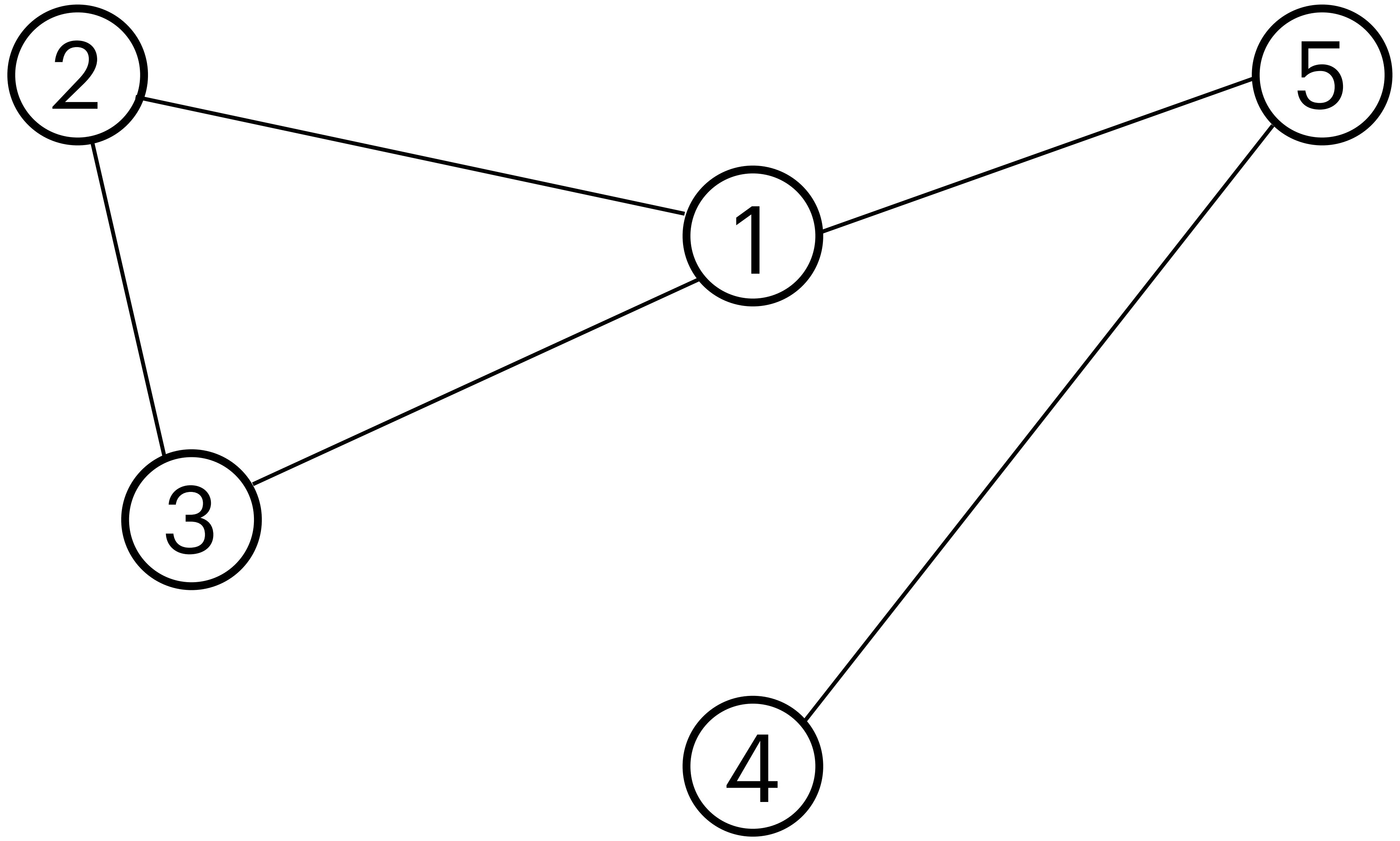


Graph

Edge

$G = (V, E)$

$E = \{e_1, ..., e_m\}, |E| = m$ the set of edges

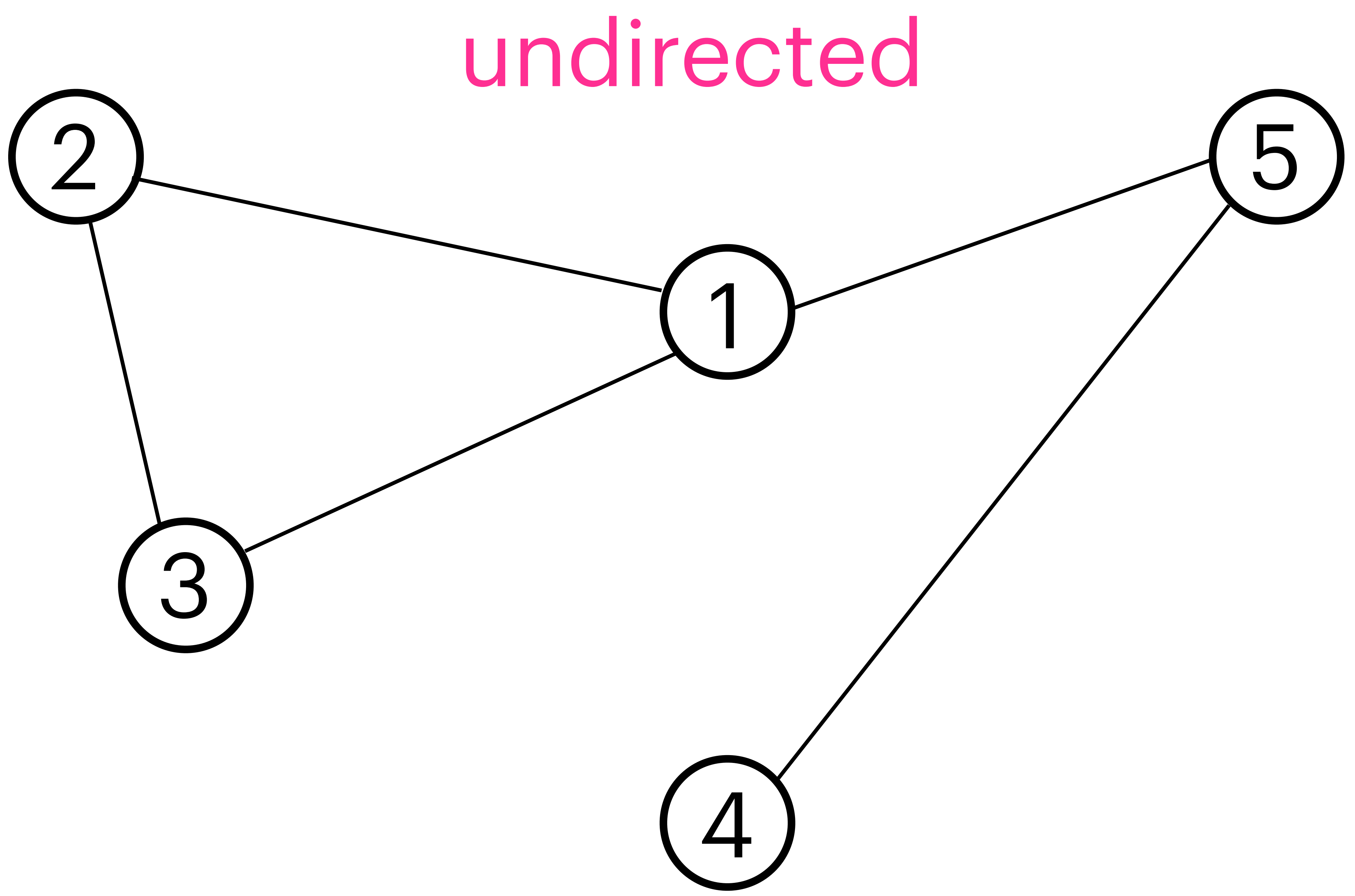


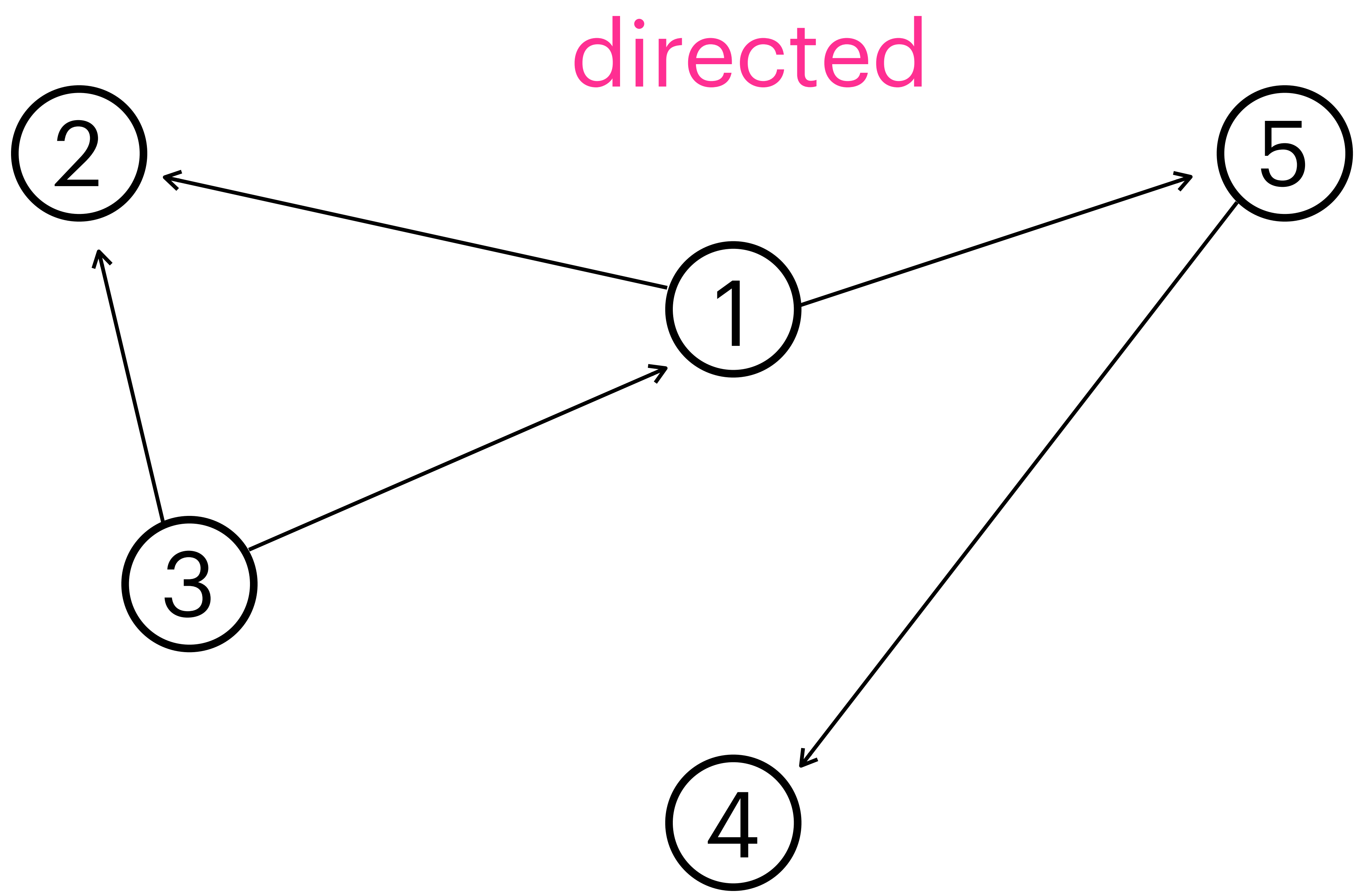
Graph

Undirected / Directed Graph

$$G = (V, E)$$

$E = \{e_1, ..., e_m\}, |E| = m$ the set of edges





$$E \subseteq \{\{u, v\} \mid u, v \in V\}$$

$$e_k = \{v_i, v_j\}$$

$$E \subseteq V \times V$$

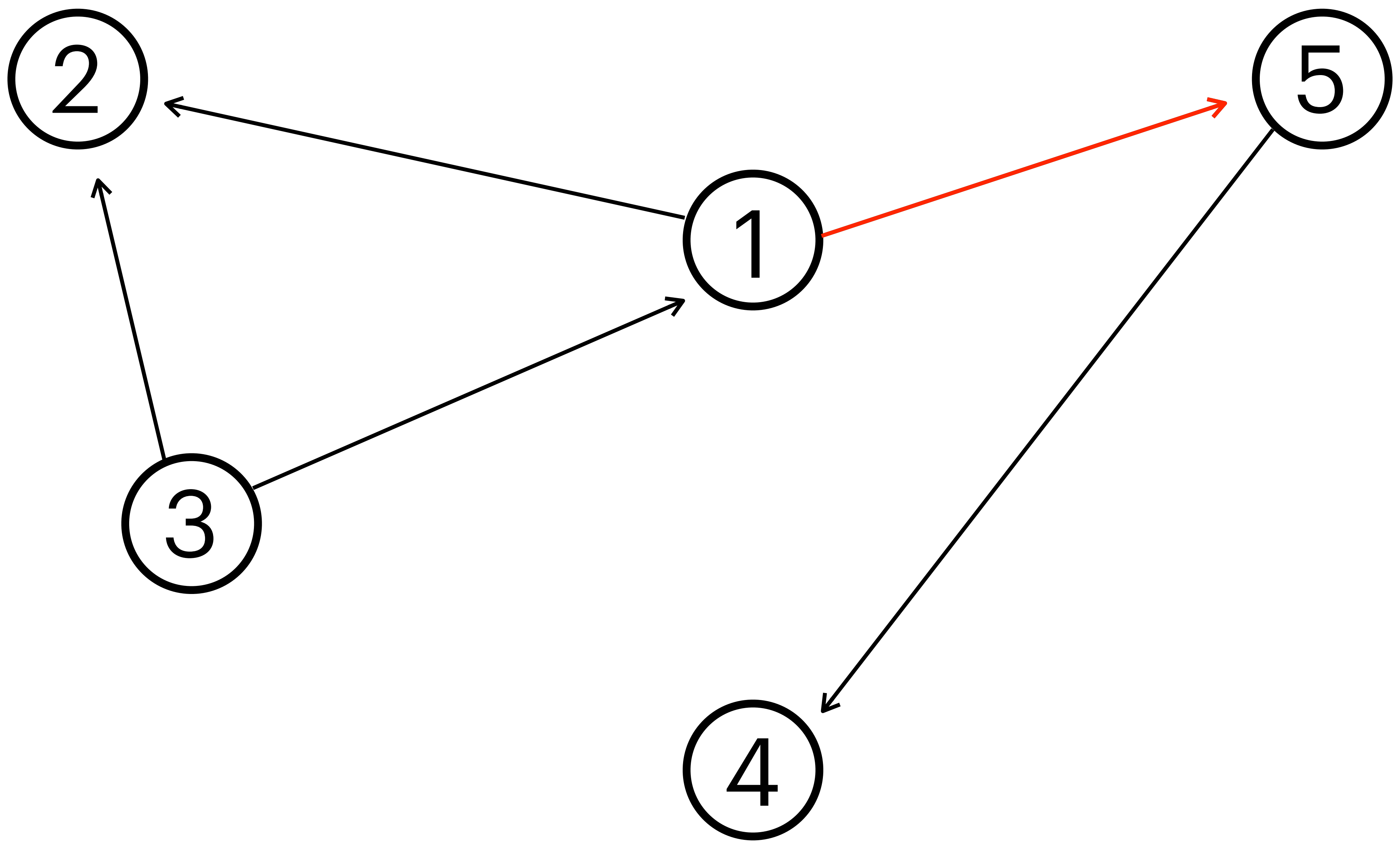
$$e_k = (v_i, v_j)$$

Graph

adjacent / incident

- v_i and v_j are adjacent iff : $\{v_i, v_j\} \in E$; $(v_i, v_j) \in E$;

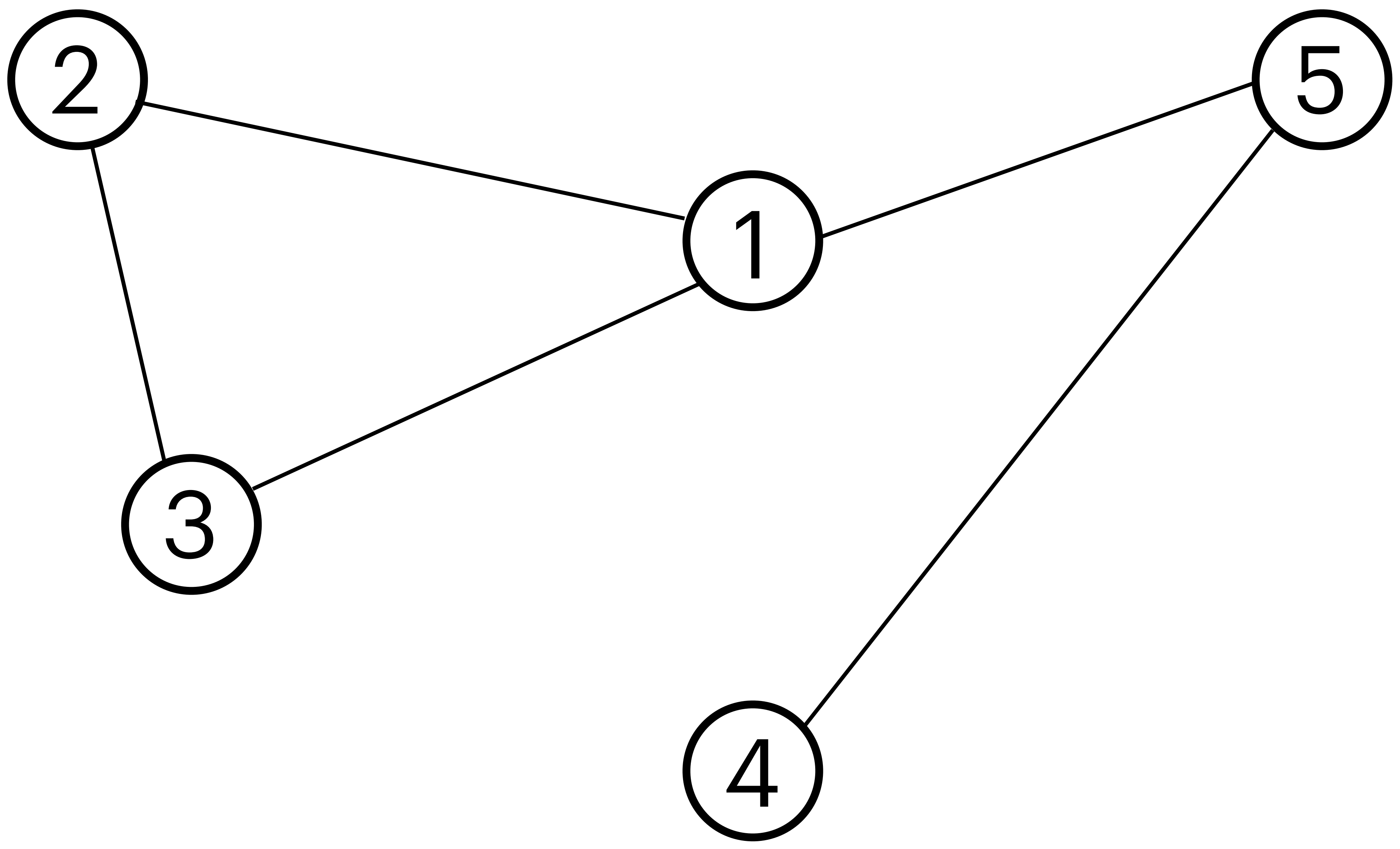
- v_i and e_k are incident iff : $v_i \in e_k$.



Graph

Degree

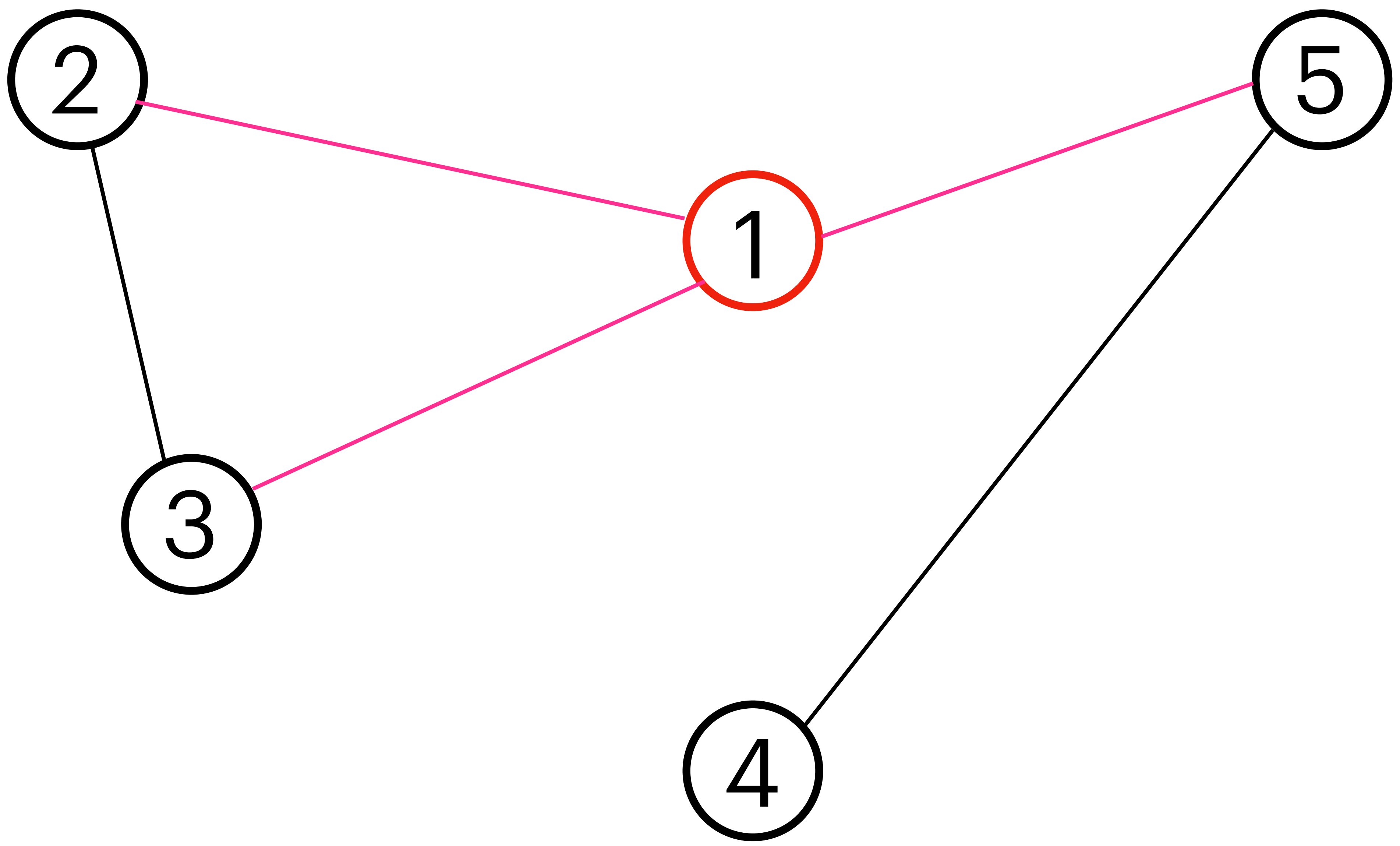
- deg(v) : The degree of a vertex v of a graph is the number of edges incident to this vertex, with loops counted twice.
- In directed graphs:
  - in-degree $deg^{-}(v)$
  - out-degree $deg^{+}(v)$



Graph

Degree

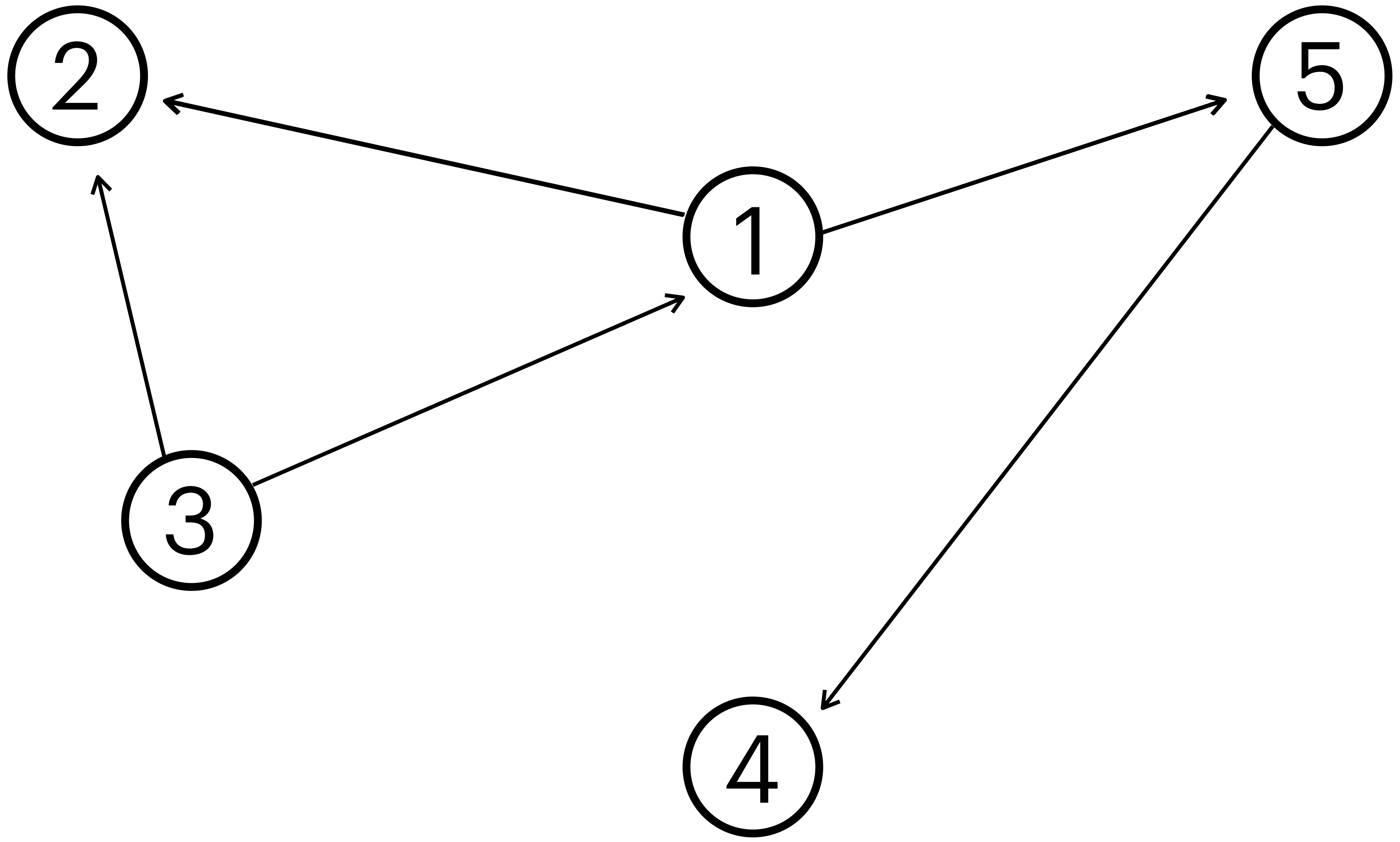
- deg(v) : The degree of a vertex v of a graph is the number of edges incident to this vertex, with loops counted twice.
- In directed graphs:
  - in-degree $deg^{-}(v)$
  - out-degree $deg^{+}(v)$



Graph

Degree

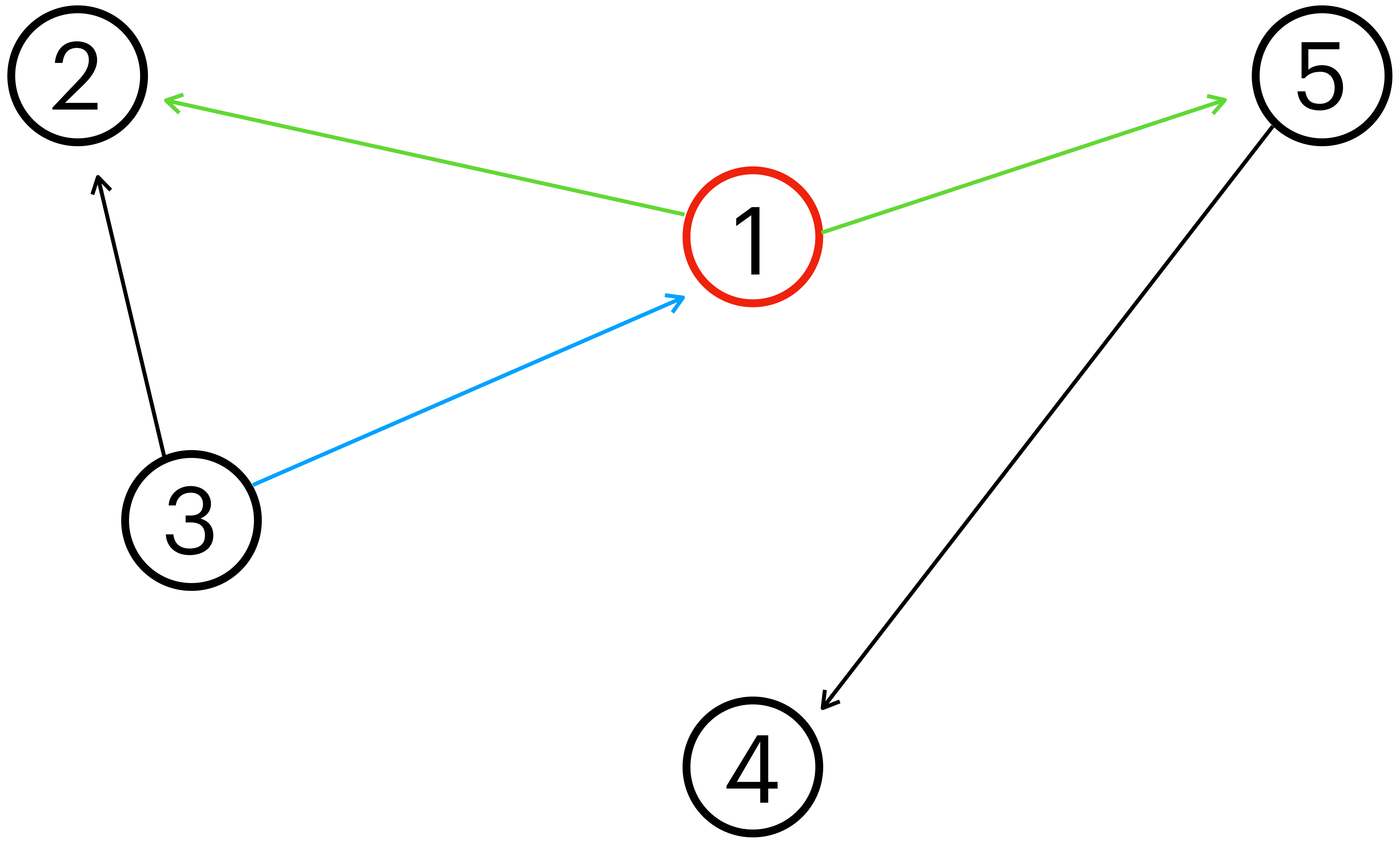
- deg(v) : The degree of a vertex v of a graph is the number of edges incident to this vertex, with loops counted twice.
- In directed graphs:
  - in-degree $deg^{-}(v)$
  - out-degree $deg^{+}(v)$



Graph

Degree

- deg(v) : The degree of a vertex v of a graph is the number of edges incident to this vertex, with loops counted twice.
- In directed graphs:
  - in-degree $deg^-(v)$
  - out-degree $deg^+(v)$



Graph
Walk vs Path

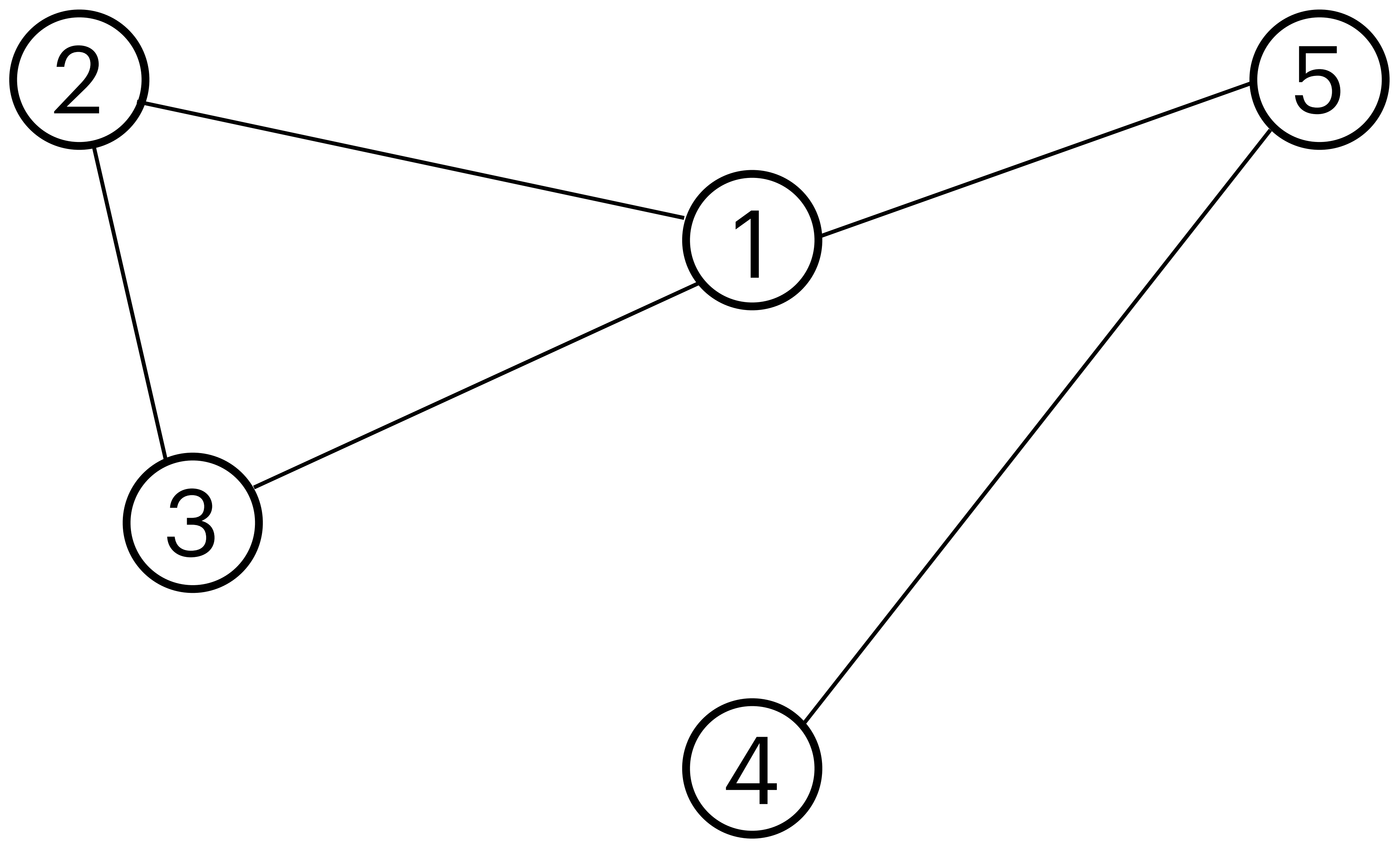
walk

- A sequence of vertices $(v_0, v_1, \ldots, v_k)$ (with $v_i \in V$ for all i) is a **walk** (german "Weg") if $\{v_i, v_{i+1}\}$ is an edge for each $0 \leq i \leq k-1$. We say that v_0 and v_k are the **endpoints** (german "Startknoten" and "Endknoten") of the walk. The **length** of the walk $(v_0, v_1, \ldots, v_k)$ is k.

path

- A sequence of vertices $(v_0, v_1, \ldots, v_k)$ is a **path** (german "Pfad") if it is a walk and all vertices are distinct (i.e., $v_i \neq v_j$ for $0 \leq i < j \leq k$).

	Is it a walk?	Is it a path?
(5, 1, 3, 2, 1)	✅	❌
(5, 1, 3)	✅	✅



Graph

Closed Walk vs Cycle

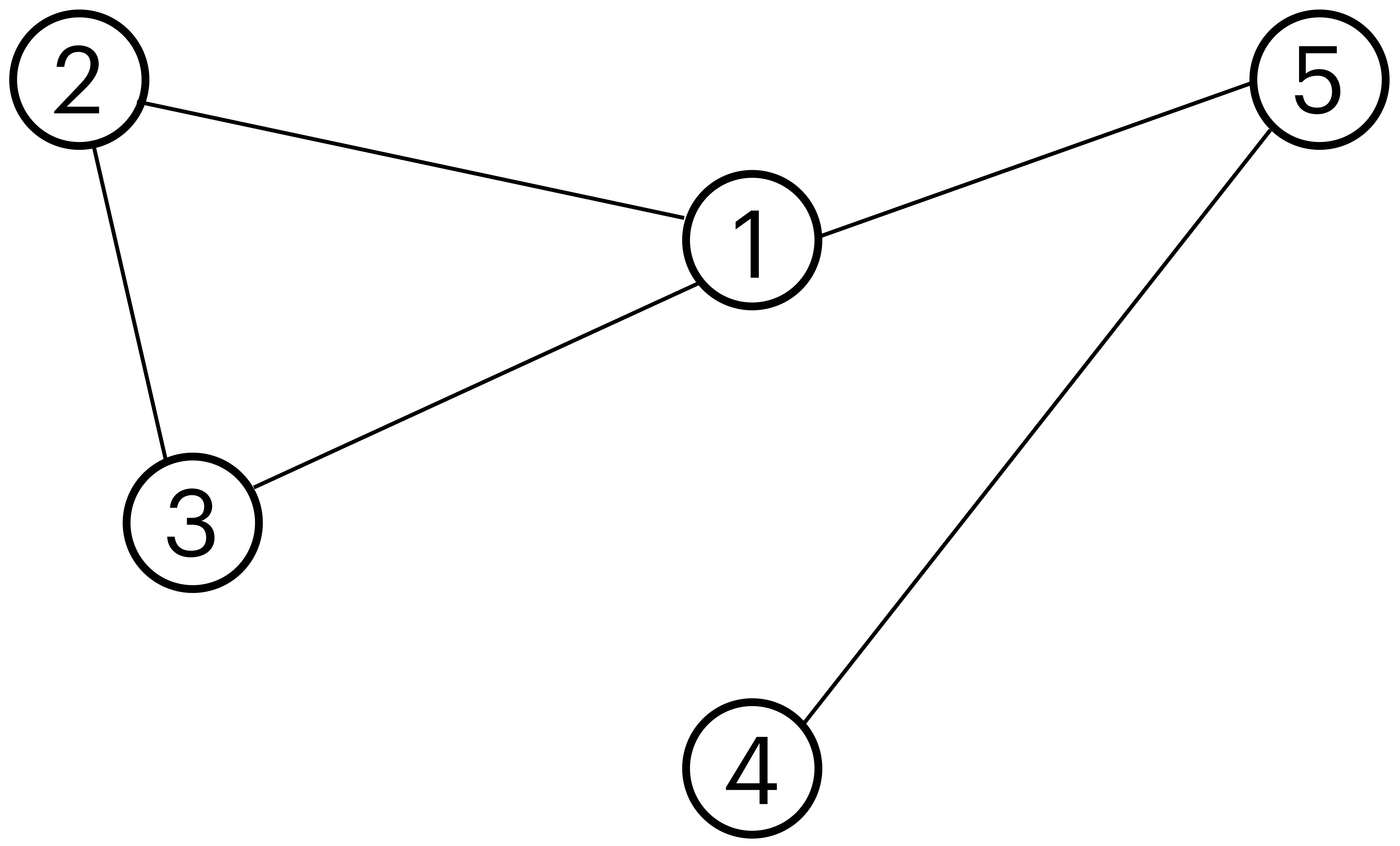
Closed walk

- A sequence of vertices $(v_0, v_1, \ldots, v_k)$ is a **closed walk** (german "Zyklus") if it is a walk, $k \geq 2$ and $v_0 = v_k$.

Cycle

- A sequence of vertices $(v_0, v_1, \ldots, v_k)$ is a **cycle** (german "Kreis") if it is a closed walk, $k \geq 3$ and all vertices (except v_0 and v_k) are distinct.

	Is it a closed walk?	Is it a cycle?
(5, 1, 3, 1, 5)	✅	❌
(1, 3, 2, 1)	✅	✅



- A sequence of vertices $(v_0, v_1, \ldots, v_k)$ (with $v_i \in V$ for all i) is a **walk** (german "Weg") if $\{v_i, v_{i+1}\}$ is an edge for each $0 \le i \le k-1$. We say that v_0 and v_k are the **endpoints** (german "Startknoten" and "Endknoten") of the walk. The **length** of the walk $(v_0, v_1, \ldots, v_k)$ is k.
- A sequence of vertices $(v_0, v_1, \ldots, v_k)$ is a **closed walk** (german "Zyklus") if it is a walk, $k \ge 2$ and $v_0 = v_k$.

Graph

Eulerian Walk / Closed Eulerian Walk

Eulerian Walk

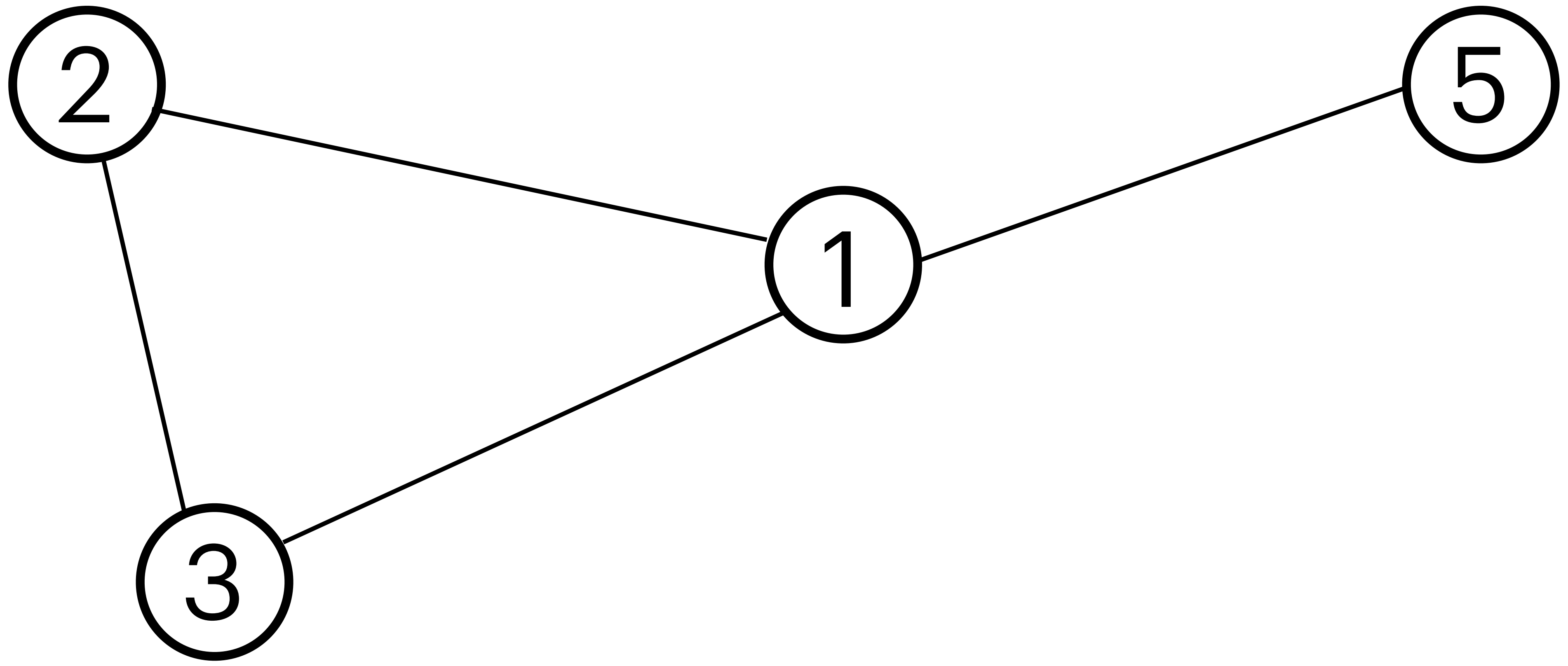
- A **Eulerian walk** (german "Eulerweg") is a walk that contains every edge exactly once.

Closed Eulerian Walk

- A **closed Eulerian walk** (german "Eulerzyklus") is a closed walk that contains every edge exactly once.

Is there an eulerian walk ? ✅ (1, 2, 3, 1, 5)

Is there a closed eulerian walk ? ❌



- A sequence of vertices $(v_0, v_1, \ldots, v_k)$ (with $v_i \in V$ for all i) is a **walk** (german "Weg") if $\{v_i, v_{i+1}\}$ is an edge for each $0 \leq i \leq k-1$. We say that v_0 and v_k are the **endpoints** (german "Startknoten" and "Endknoten") of the walk. The **length** of the walk $(v_0, v_1, \ldots, v_k)$ is k.

- A sequence of vertices $(v_0, v_1, \ldots, v_k)$ is a **closed walk** (german "Zyklus") if it is a walk, $k \geq 2$ and $v_0 = v_k$.

Graph

Eulerian Walk / Closed Eulerian Walk

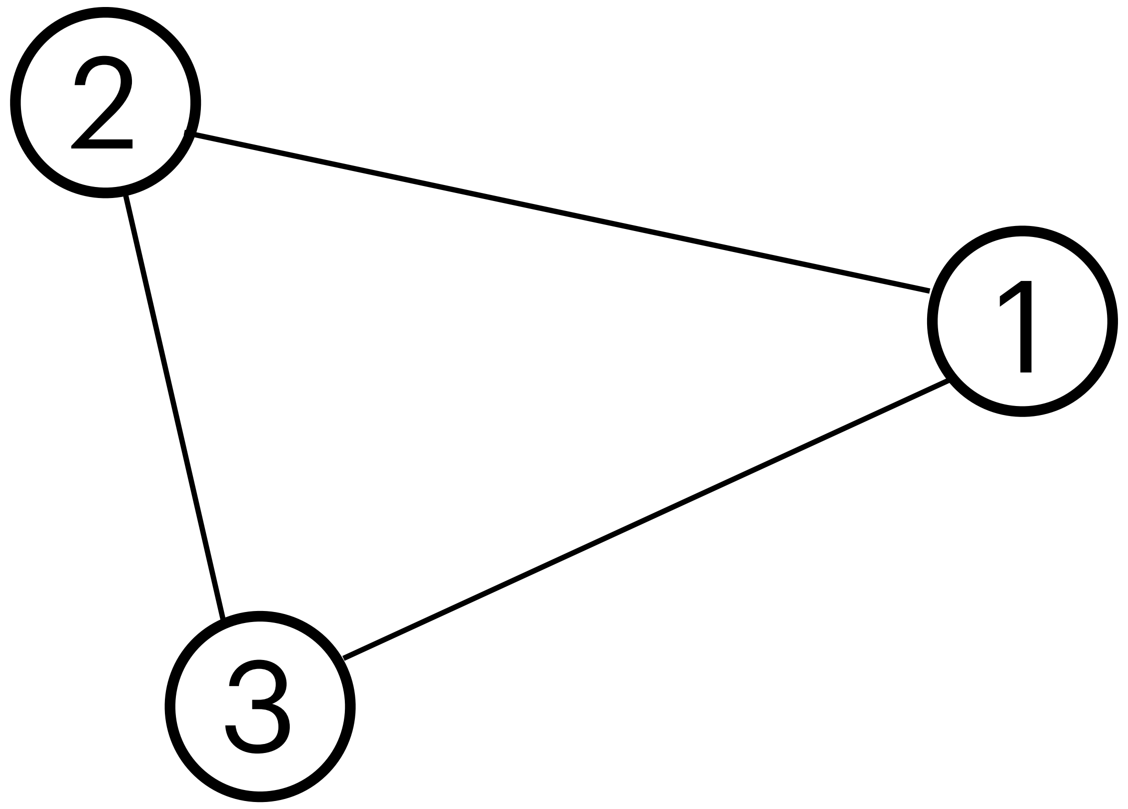
Eulerian Walk

- A **Eulerian walk** (german "Eulerweg") is a walk that contains every edge exactly once.

Closed Eulerian Walk

- A **closed Eulerian walk** (german "Eulerzyklus") is a closed walk that contains every edge exactly once.

Is there a closed eulerian walk ? ✅ (1, 2, 3, 1)



Graph

Hamiltonian Path / Hamiltonian Cycle

- A sequence of vertices $(v_0, v_1, \ldots, v_k)$ is a **path** (german "Pfad") if it is a walk and all vertices are distinct (i.e., $v_i \neq v_j$ for $0 \leq i < j \leq k$).
- A sequence of vertices $(v_0, v_1, \ldots, v_k)$ is a **cycle** (german "Kreis") if it is a closed walk, $k \geq 3$ and all vertices (except v_0 and v_k) are distinct.

Hamiltonian Path

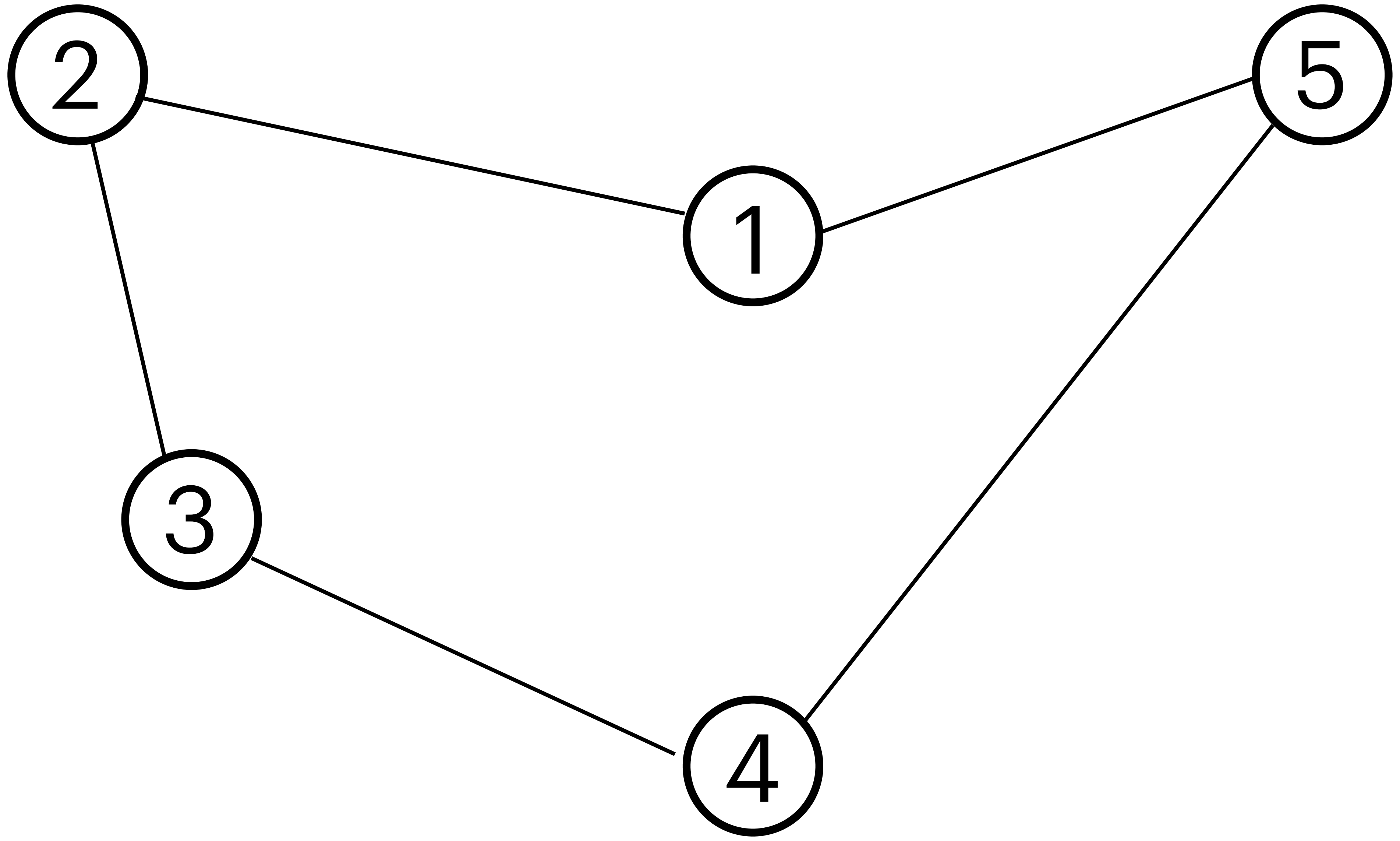
- A **Hamiltonian path** (german "Hamiltonpfad") is a path that contains every vertex.

Hamiltonian Cycle

- A **Hamiltonian cycle** (german "Hamiltonkreis") is a cycle that contains every vertex.

Is it a hamiltonian path? Is it a hamiltonian cycle?

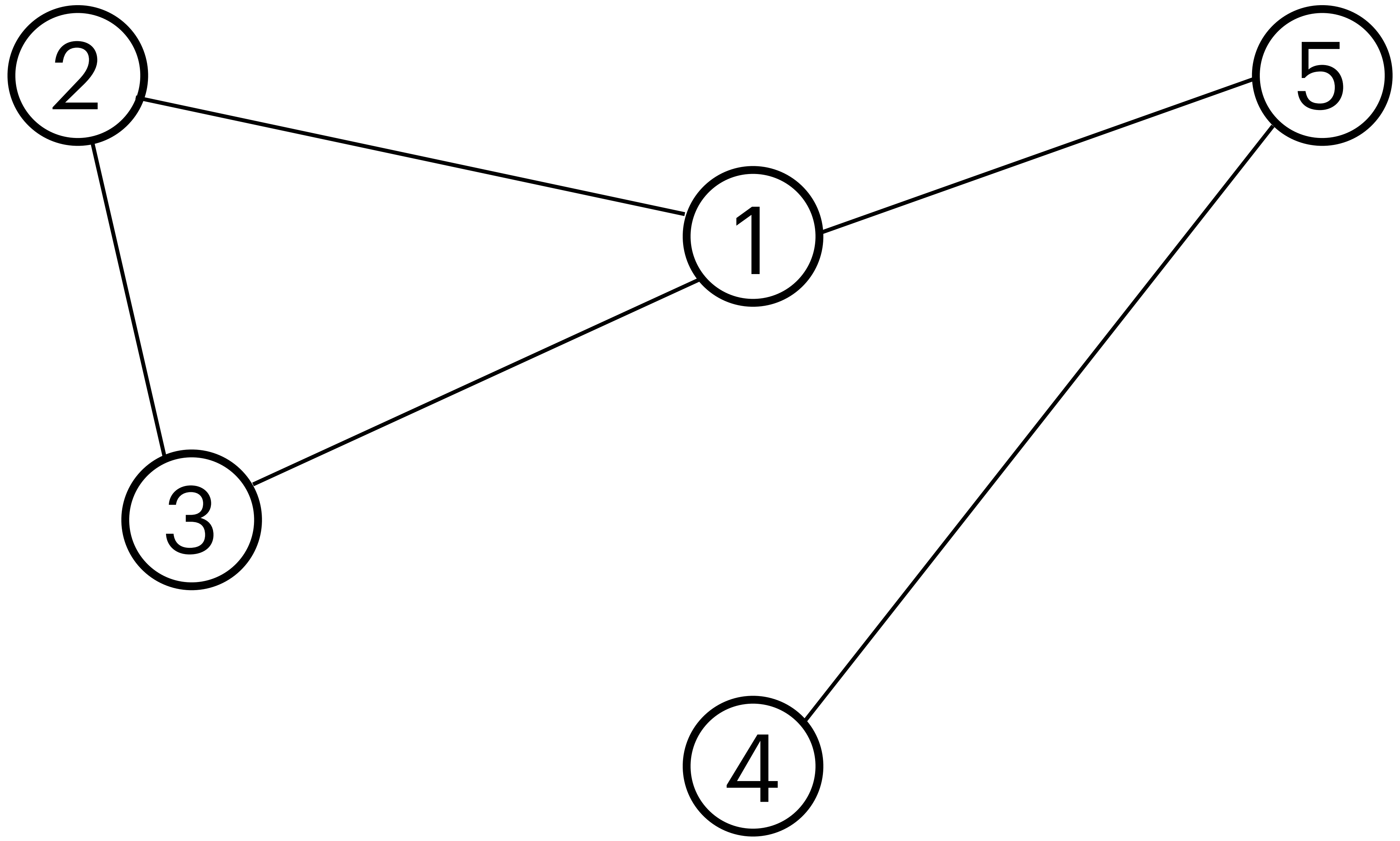
(5, 1, 2, 3, 4)	✅	❌
(5, 1, 2, 3, 4, 5)	❌	✅



Graph

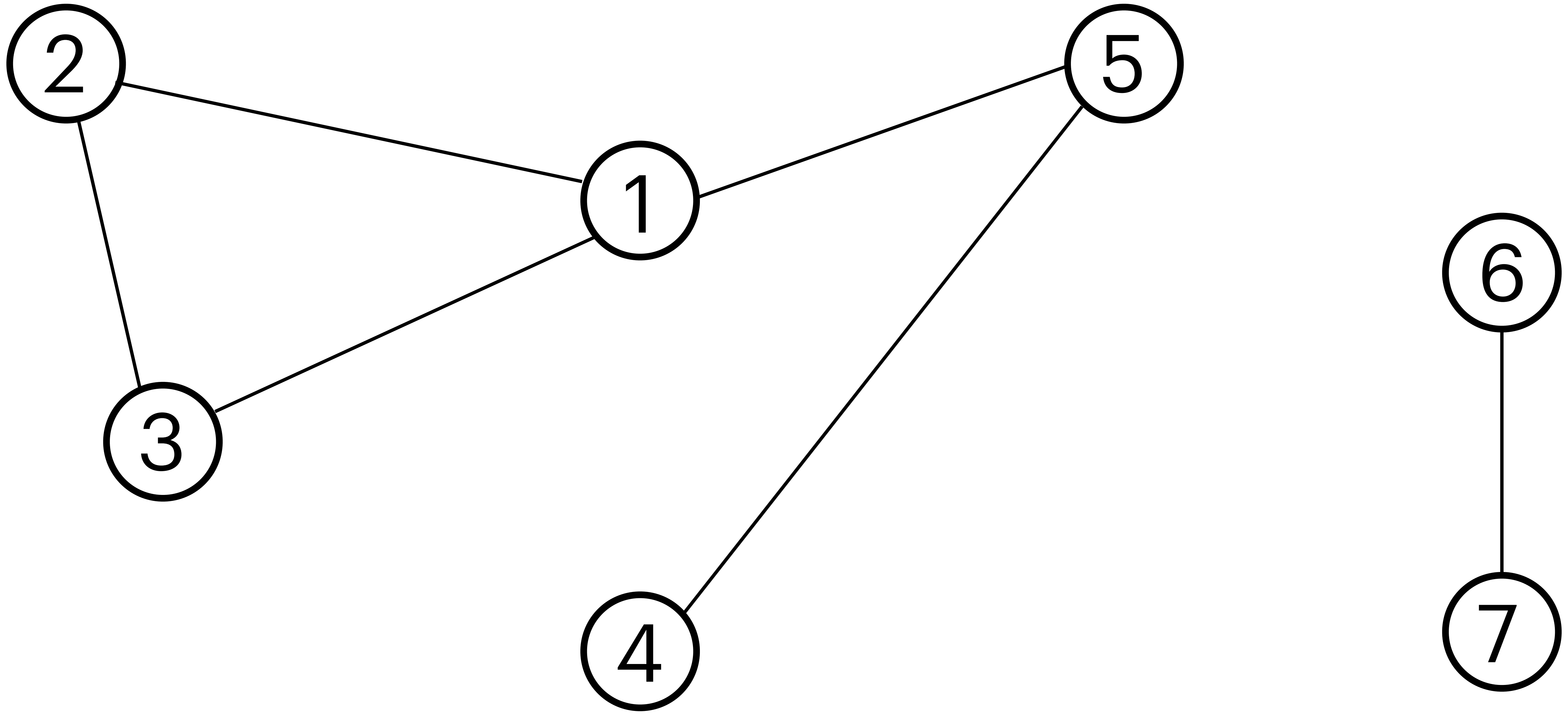
Connectivity

- For $u, v \in V$, we say u **reaches** v (or v is **reachable** from u; german "u erreicht v") if there exists a walk with endpoints u and v.
- A **connected component** of G is an equivalence class of the (equivalence) relation defined as follows: Two vertices $u, v \in V$ are equivalent if u reaches v.
- A graph G is **connected** (german "zusammenhängend") if for every two vertices $u, v \in V$ u reaches v or equivalently if there is only one connected component.



Graph
Connectivity

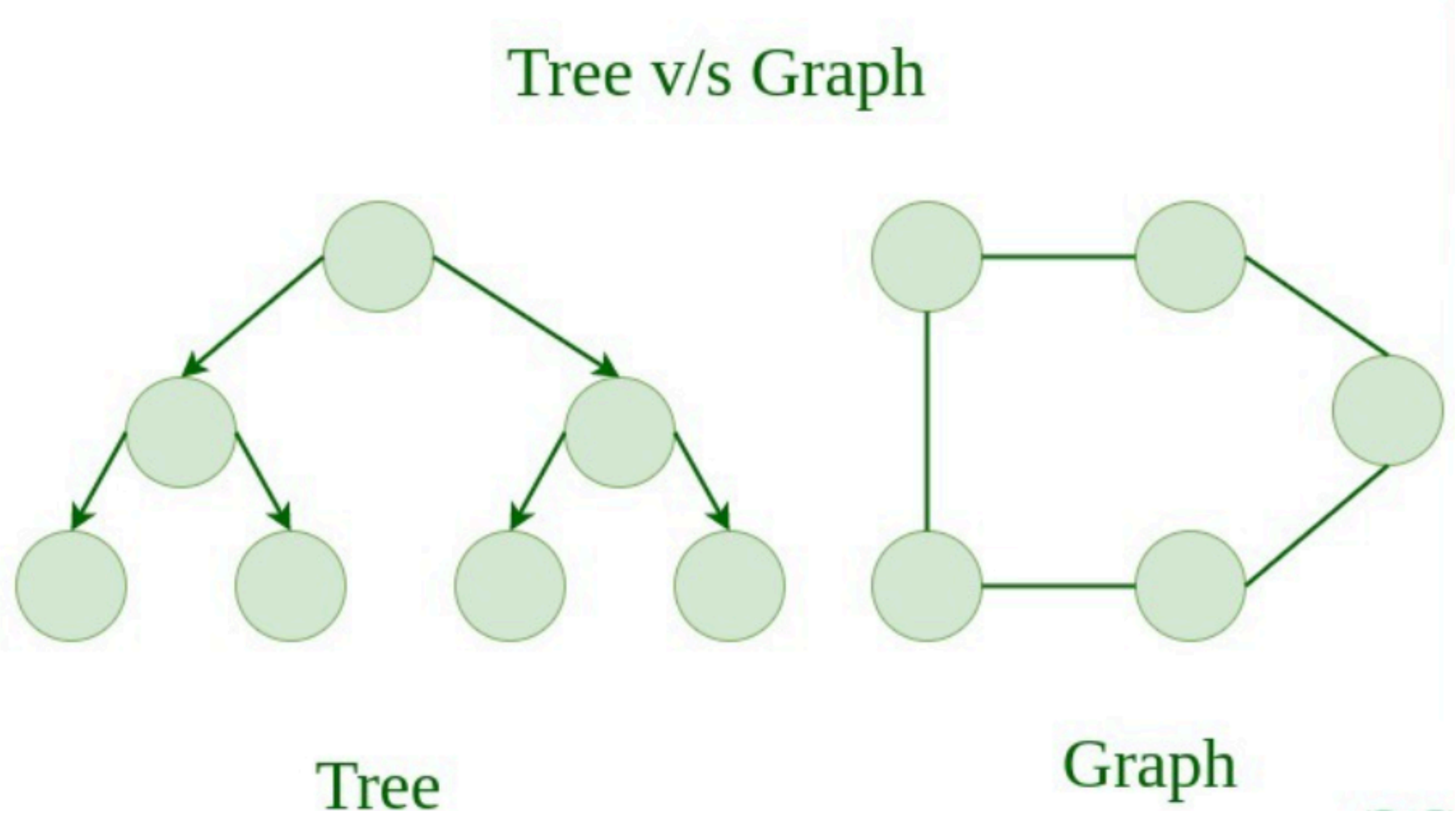
- For $u, v \in V$, we say u **reaches** v (or v is **reachable** from u; german "u erreicht v") if there exists a walk with endpoints u and v.
- A **connected component** of G is an equivalence class of the (equivalence) relation defined as follows: Two vertices $u, v \in V$ are equivalent if u reaches v.
- A graph G is **connected** (german "zusammenhängend") if for every two vertices $u, v \in V$ u reaches v or equivalently if there is only one connected component.



Graph
Tree

- A graph G is a **tree** (german "Baum") if it is connected and has no cycles.

A connected graph is a tree iff it has exactly m= n−1 edges.



Graph

Handshaking Lemma

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

Graph
Eulerian Lemmas

- A **Eulerian walk** (german "Eulerweg") is a walk that contains every edge exactly once.
- A **closed Eulerian walk** (german "Eulerzyklus") is a closed walk that contains every edge exactly once.

$\exists$ Eulerian Walk $\Longleftrightarrow$ <= 2 vertices have odd degree (start and end)

Ausser für Start- und Endknoten gilt : #hin = #weg

$\exists$ Eulerian Closed Walk $\Longleftrightarrow$ G is connected and every vertex has even degree

Graph
Definitions

Definition 1. Let $G = (V, E)$ be a graph.

- For $v \in V$, the **degree** $\deg(v)$ of v (german "Knotengrad") is the number of edges that are incident to v.
- A sequence of vertices $(v_0, v_1, \ldots, v_k)$ (with $v_i \in V$ for all i) is a **walk** (german "Weg") if $\{v_i, v_{i+1}\}$ is an edge for each $0 \leq i \leq k-1$. We say that v_0 and v_k are the **endpoints** (german "Startknoten" and "Endknoten") of the walk. The **length** of the walk $(v_0, v_1, \ldots, v_k)$ is k.
- A sequence of vertices $(v_0, v_1, \ldots, v_k)$ is a **closed walk** (german "Zyklus") if it is a walk, $k \geq 2$ and $v_0 = v_k$.
- A sequence of vertices $(v_0, v_1, \ldots, v_k)$ is a **path** (german "Pfad") if it is a walk and all vertices are distinct (i.e., $v_i \neq v_j$ for $0 \leq i < j \leq k$).
- A sequence of vertices $(v_0, v_1, \ldots, v_k)$ is a **cycle** (german "Kreis") if it is a closed walk, $k \geq 3$ and all vertices (except v_0 and v_k) are distinct.
- A **Eulerian walk** (german "Eulerweg") is a walk that contains every edge exactly once.
- A **closed Eulerian walk** (german "Eulerzyklus") is a closed walk that contains every edge exactly once.
- A **Hamiltonian path** (german "Hamiltonpfad") is a path that contains every vertex.
- A **Hamiltonian cycle** (german "Hamiltonkreis") is a cycle that contains every vertex.
- For $u, v \in V$, we say u **reaches** v (or v is **reachable** from u; german "u erreicht v") if there exists a walk with endpoints u and v.
- A **connected component** of G is an equivalence class of the (equivalence) relation defined as follows: Two vertices $u, v \in V$ are equivalent if u reaches v.
- A graph G is **connected** (german "zusammenhängend") if for every two vertices $u, v \in V$ u reaches v or equivalently if there is only one connected component.
- A graph G is a **tree** (german "Baum") if it is connected and has no cycles.

Let’s take a break

Graph
Short Proofs

Exercise 8.5 *Short questions about graphs* **(2 points).**

In the following, let $G = (V, E)$ be a graph, $n = |V|$ and $m = |E|$.

(a) Let $v \neq w \in V$. Prove that if there is a walk with endpoints v and w, then there is a path with endpoints v and w.

For each of the following statements, decide whether the statement is true or false. If the statement is true, provide a proof; if it is false, provide a counterexample.

(b) Every graph with $m \geq n$ is connected.

(c) If G contains a Hamiltonian path, then G contains a Eulerian walk.

(d) If every vertex of a non-empty graph G has degree at least 2, then G contains a cycle.

(e) Suppose in a graph G every pair of vertices v, w has a common neighbour (i.e., for all distinct vertices v, w, there is a vertex x such that $\{v, x\}$ and $\{w, x\}$ are both edges). Then there exists a vertex p in G which is a neighbour of every other vertex in G (i.e., p has degree $n - 1$).

(f) Let G be a connected graph with at least 3 vertices. Suppose there exists a vertex v_{cut} in G so that after deleting v_{cut}, G is no longer connected. Then G does not have a Hamiltonian cycle. (Deleting a vertex v means that we remove v and any edge containing v from the graph).

Graph
Exam Question

/ 5 P

c) *Graph quiz:* For each of the following claims, state whether it is true or false. You get 1P for a correct answer, -1P for a wrong answer, 0P for a missing answer. You get at least 0 points in total.

As a reminder, here are a few definitions for a (directed) graph $G = (V, E)$:

For $k \geq 2$, a (directed) *walk* is a sequence of vertices $v_1, \ldots, v_k$ such that for every two consecutive vertices v_i, v_{i+1}, we have $\{v_i, v_{i+1}\} \in E$ (resp. $(v_i, v_{i+1}) \in E$ for a directed walk).

A (directed) *closed walk* is a (directed) walk with $v_1 = v_k$.

A (directed) *cycle* is a (directed) closed walk where $k \geq 3$ and all vertices (except v_1 and v_k) are distinct.

A (directed) *closed Eulerian walk* is a (directed) closed walk which traverses every edge in E exactly once.

For a vertex v in a directed graph $G = (V, E)$, the *in-degree* of v is the number of edges in E that end in v (i.e., of the form (w, v)), and the *out-degree* of v is the number of edges in E that start in v (i.e., of the form (v, w)).

Claim	**true**	**false**
A connected graph must contain a cycle.	☐	☐
A graph $G = (V, E)$ with $\lvert E\rvert \leq \lvert V\rvert - 1$ is a tree.	☐	☐
Let $G = (V, E)$ be a graph with $\lvert E\rvert \geq 4$, which contains a closed Eulerian walk. If we remove one edge from E, the resulting graph does not contain a closed Eulerian walk, no matter which edge we remove (the vertex set does not change).	☐	☐
Let $G = (V, E)$ be a *directed* graph. If the in-degree and out-degree of every vertex $v \in V$ is even, then G contains a *directed* closed Eulerian walk.	☐	☐
Let $G = (V, E)$ be an undirected graph. Then there is a way to direct the edges of G such that the resulting directed graph does not contain a directed cycle.	☐	☐

Graph Definitions
Exam Tipps

- T/F or a proof !
- Know all of the definitions
  - Don’t mix up similar ones, realise connections!
    - walk , closed walk , eulerian
    - path , cycle, hamiltonian
- Gain an intuition
  - Don’t rush ! Don’t gamble !
  - Do this by actually coming up with short proofs !
- Practice !

Next Week

Questions

Feedbacks , Recommendations

Nil Ozer

Additional Slides

Euler(G)

Euler(G): (finde Eulerzyklus in G wenn existent)

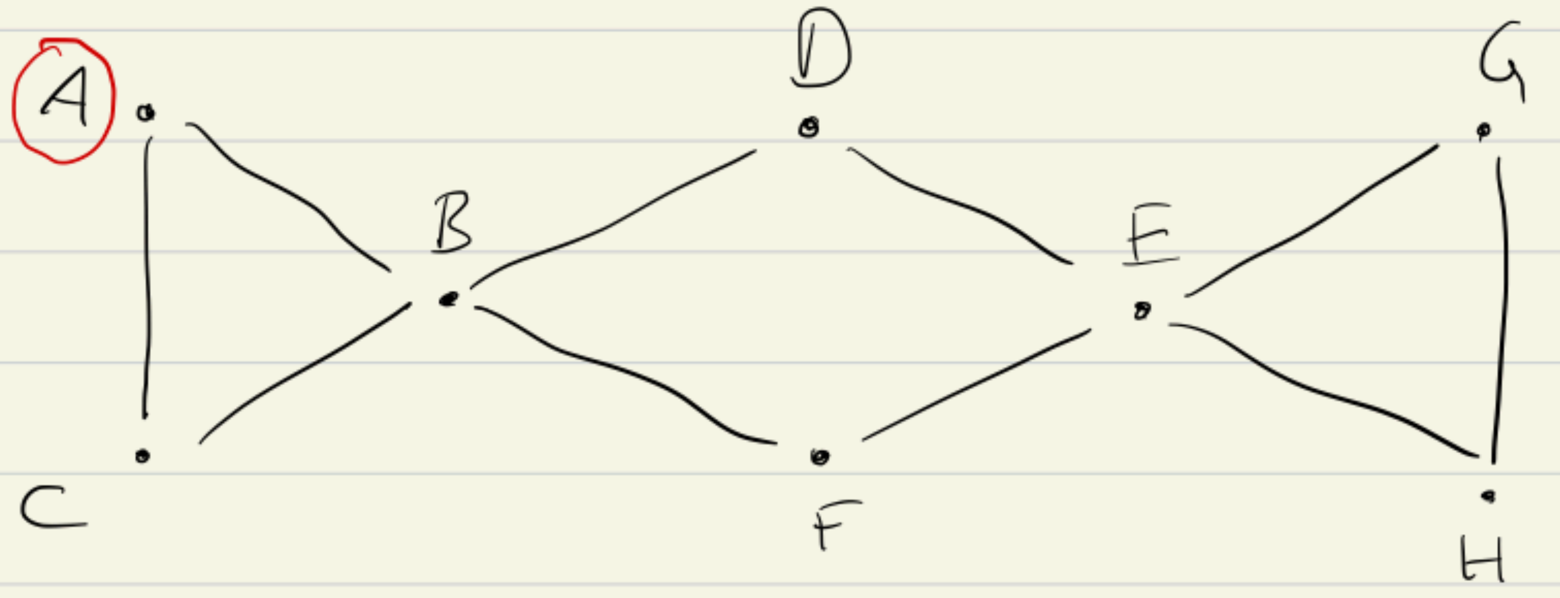
- Leere Liste Z, alle Kanten unmarkiert
- EulerWalk(u_0) für $u_0 \in V$ beliebig
- return Z

EulerWalk(u): ⚠ auch nicht in Rekursion markiert!

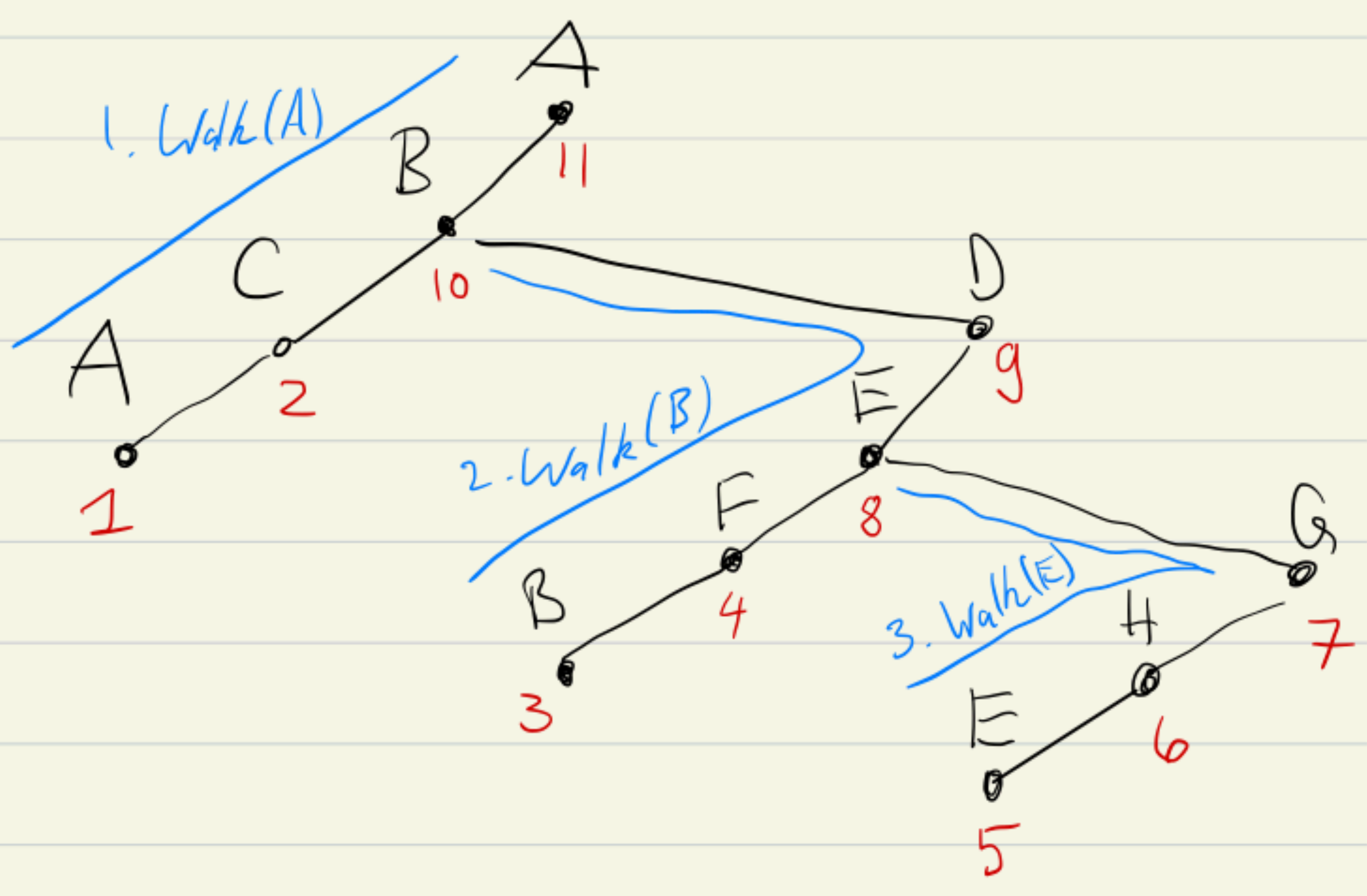
for $uv \in E$, nicht markiert
  markiere Kante uv
  EulerWalk(v)

$Z \leftarrow (Z, u)$

Beispiel:



Rekursionsbaum



Z: A C B F E H G E D B A