Basic graph definitions

In a simple graph, we will write $e = uv$ to mean that e is the edge with endpoints u and v.

Basic graph definitions

In a simple graph, we will write $e = uv$ to mean that e is the edge with endpoints u and v.

We then say that u and v are **adjacent**, or that they are **neighbors**.

Basic graph definitions

In a simple graph, we will write $e = uv$ to mean that e is the edge with endpoints u and v.

We then say that u and v are **adjacent**, or that they are **neighbors**.

We also say that e is **incident** to u and v.

Basic graph definitions

In a simple graph, we will write $e = uv$ to mean that e is the edge with endpoints u and v.

We then say that u and v are **adjacent**, or that they are **neighbors**.

We also say that e is **incident** to u and v.

The **order** or a graph is the number of vertices, usually denoted by $n = |V(G)|$.

Basic graph definitions

In a simple graph, we will write $e = uv$ to mean that e is the edge with endpoints u and v.

We then say that u and v are **adjacent**, or that they are **neighbors**.

We also say that e is **incident** to u and v.

The **order** or a graph is the number of vertices, usually denoted by $n = |V(G)|$.

The **size** or a graph is the number of edges, usually denoted by $e = |E(G)|$.

More definitions

The **complement** of a simple graph G, denoted by $\overline{G}$, is the simple graph with the same vertex set as G, and with edges set $E(\overline{G})$ defined by

$$uv \in E(\overline{G}) \iff uv \notin E(G).$$

More definitions

The **complement** of a simple graph G, denoted by $\overline{G}$, is the simple graph with the same vertex set as G, and with edges set $E(\overline{G})$ defined by

$$uv \in E(\overline{G}) \iff uv \notin E(G).$$

A **clique** in a simple graph is a set of pairwise adjacent vertices.

More definitions

The **complement** of a simple graph G, denoted by $\overline{G}$, is the simple graph with the same vertex set as G, and with edges set $E(\overline{G})$ defined by

$$uv \in E(\overline{G}) \iff uv \notin E(G).$$

A **clique** in a simple graph is a set of pairwise adjacent vertices.

An **independent set** in a graph is a set of pairwise non-adjacent vertices.

More definitions

The **complement** of a simple graph G, denoted by $\overline{G}$, is the simple graph with the same vertex set as G, and with edges set $E(\overline{G})$ defined by

$$uv \in E(\overline{G}) \iff uv \notin E(G).$$

A **clique** in a simple graph is a set of pairwise adjacent vertices.

An **independent set** in a graph is a set of pairwise non-adjacent vertices.

A graph G is **bipartite** if $V(G)$ is the union of two disjoint (possibly empty) independent sets. In other words, we can partition $V(G) = V_1 \sqcup V_2$ so that all edges go between V_1 and V_2.

Coloring and planarity

Definition

The **chromatic number** of a graph G, denoted by $\chi(G)$, is the minimum number of colors needed to color the vertices so that adjacent vertices receive different colors.

Coloring and planarity

Definition

The **chromatic number** of a graph G, denoted by $\chi(G)$, is the minimum number of colors needed to color the vertices so that adjacent vertices receive different colors.

- Vertices in a clique must all receive different colors.

Coloring and planarity

Definition

The **chromatic number** of a graph G, denoted by $\chi(G)$, is the minimum number of colors needed to color the vertices so that adjacent vertices receive different colors.

- Vertices in a clique must all receive different colors.
- Vertices of the same color form an independent set.

Coloring and planarity

Definition

The **chromatic number** of a graph G, denoted by $\chi(G)$, is the minimum number of colors needed to color the vertices so that adjacent vertices receive different colors.

- Vertices in a clique must all receive different colors.
- Vertices of the same color form an independent set.
- $\chi(G)$ is the minimum number of independent sets needed to partition $V(G)$.

Coloring and planarity

Definition

The **chromatic number** of a graph G, denoted by $\chi(G)$, is the minimum number of colors needed to color the vertices so that adjacent vertices receive different colors.

- Vertices in a clique must all receive different colors.
- Vertices of the same color form an independent set.
- $\chi(G)$ is the minimum number of independent sets needed to partition $V(G)$.
  In particular, G is bipartite if and only if $\chi(G) \leq 2$.

Coloring and planarity

Definition

The **chromatic number** of a graph G, denoted by $\chi(G)$, is the minimum number of colors needed to color the vertices so that adjacent vertices receive different colors.

- Vertices in a clique must all receive different colors.
- Vertices of the same color form an independent set.
- $\chi(G)$ is the minimum number of independent sets needed to partition $V(G)$.
  In particular, G is bipartite if and only if $\chi(G) \leq 2$.

Definition

A graph is **planar** if it can be drawn on the plane without crossing edges.

A few special graphs

- P_n is the *path* on n vertices.

A few special graphs

- P_n is the *path* on n vertices.
- C_n is the *cycle* on n vertices.

A few special graphs

- P_n is the *path* on n vertices.
- C_n is the *cycle* on n vertices.
- K_n is the complete graph on n vertices.

A few special graphs

- P_n is the *path* on n vertices.
- C_n is the *cycle* on n vertices.
- K_n is the complete graph on n vertices.
- $K_{m,n}$ is the complete bipartite graph with partite sets of sizes m and n.

Subgraphs

Definition

A **subgraph** of G is a graph H that can be obtained from G by deleting vertices and/or edges. We write $H \subseteq G$.

Adjacency and Incidence matrices

Let G be a graph with n vertices and m edges.

Definition

The **adjacency matrix** $A(G)$ is the $n \times n$ matrix in which

$$a_{ij} = \#\text{edges in } G \text{ with endpoints } v_i \text{ and } v_j.$$

Adjacency and Incidence matrices

Let G be a graph with n vertices and m edges.

Definition

The **adjacency matrix** $A(G)$ is the $n \times n$ matrix in which

$$a_{ij} = \#\text{edges in } G \text{ with endpoints } v_i \text{ and } v_j.$$

Question: if G is a simple graph, what can the entries of $A(G)$ be?

Adjacency and Incidence matrices

Let G be a graph with n vertices and m edges.

Definition

The **adjacency matrix** $A(G)$ is the $n \times n$ matrix in which

$$a_{ij} = \#\text{edges in } G \text{ with endpoints } v_i \text{ and } v_j.$$

Question: if G is a simple graph, what can the entries of $A(G)$ be?

Definition

The **incidence matrix** $M(G)$ is the $n \times m$ matrix in which

$$m_{ij} = \begin{cases} 1 & \text{if } v_i \text{ is an endpoint of } e_j, \\ 0 & \text{otherwise.} \end{cases}$$

Adjacency and Incidence matrices

Let G be a graph with n vertices and m edges.

Definition

The **adjacency matrix** $A(G)$ is the $n \times n$ matrix in which

$$a_{ij} = \#\text{edges in } G \text{ with endpoints } v_i \text{ and } v_j.$$

Question: if G is a simple graph, what can the entries of $A(G)$ be?

Definition

The **incidence matrix** $M(G)$ is the $n \times m$ matrix in which

$$m_{ij} = \begin{cases} 1 & \text{if } v_i \text{ is an endpoint of } e_j, \\ 0 & \text{otherwise.} \end{cases}$$

Definition

The **degree** of a vertex is the number of edges incident to it.

Graph Isomorphism

Definition

An isomorphism from a simple graph G to a simple graph H is a bijection

$$f : V(G) \to V(H)$$

such that $uv \in E(G)$ if and only if $f(u)f(v) \in E(H)$.

We write $G \cong H$ to mean that G is isomorphic to H.

Graph Isomorphism

Definition

An isomorphism from a simple graph G to a simple graph H is a bijection

$$f : V(G) \to V(H)$$

such that $uv \in E(G)$ if and only if $f(u)f(v) \in E(H)$.

We write $G \cong H$ to mean that G is isomorphic to H.

Intuitively, G and H are isomorphic if they are drawings of the "same" graph, possibly with a different labeling of the vertices.

Graph Isomorphism

Definition

An isomorphism from a simple graph G to a simple graph H is a bijection

$$f : V(G) \to V(H)$$

such that $uv \in E(G)$ if and only if $f(u)f(v) \in E(H)$.

We write $G \cong H$ to mean that G is isomorphic to H.

Intuitively, G and H are isomorphic if they are drawings of the "same" graph, possibly with a different labeling of the vertices.

Isomorphism is an equivalence relation on the set of all graphs. We think of isomorphism classes as *unlabeled graphs*.

Graph Isomorphism

Definition

An isomorphism from a simple graph G to a simple graph H is a bijection

$$f : V(G) \to V(H)$$

such that $uv \in E(G)$ if and only if $f(u)f(v) \in E(H)$.

We write $G \cong H$ to mean that G is isomorphic to H.

Intuitively, G and H are isomorphic if they are drawings of the "same" graph, possibly with a different labeling of the vertices.

Isomorphism is an equivalence relation on the set of all graphs. We think of isomorphism classes as *unlabeled graphs*.

We can show that two graphs are isomorphic by constructing a bijection as above.

Graph Isomorphism

Definition

An isomorphism from a simple graph G to a simple graph H is a bijection

$$f : V(G) \to V(H)$$

such that $uv \in E(G)$ if and only if $f(u)f(v) \in E(H)$.

We write $G \cong H$ to mean that G is isomorphic to H.

Intuitively, G and H are isomorphic if they are drawings of the "same" graph, possibly with a different labeling of the vertices.

Isomorphism is an equivalence relation on the set of all graphs. We think of isomorphism classes as *unlabeled graphs*.

We can show that two graphs are isomorphic by constructing a bijection as above.

But how do we show that two graphs are not isomorphic?

Showing that two graphs are *not* isomorphic

To show two graphs are not isomorphic, we can find a “structural property” (preserved by isomorphisms) on which they differ.

Showing that two graphs are *not* isomorphic

To show two graphs are not isomorphic, we can find a "structural property" (preserved by isomorphisms) on which they differ.

For example:

- One is bipartite and the other is not.

Showing that two graphs are *not* isomorphic

To show two graphs are not isomorphic, we can find a "structural property" (preserved by isomorphisms) on which they differ.

For example:

- One is bipartite and the other is not.
- They have distinct complements.

Showing that two graphs are *not* isomorphic

To show two graphs are not isomorphic, we can find a "structural property" (preserved by isomorphisms) on which they differ.

For example:

- One is bipartite and the other is not.
- They have distinct complements.
- They have different order or size.

Showing that two graphs are *not* isomorphic

To show two graphs are not isomorphic, we can find a "structural property" (preserved by isomorphisms) on which they differ.

For example:

- One is bipartite and the other is not.
- They have distinct complements.
- They have different order or size.
- One graph has a vertex of degree k and the other doesn't.

Showing that two graphs are *not* isomorphic

To show two graphs are not isomorphic, we can find a "structural property" (preserved by isomorphisms) on which they differ.

For example:

- One is bipartite and the other is not.
- They have distinct complements.
- They have different order or size.
- One graph has a vertex of degree k and the other doesn't.
- Their chromatic number is not the same.

Showing that two graphs are *not* isomorphic

To show two graphs are not isomorphic, we can find a "structural property" (preserved by isomorphisms) on which they differ.

For example:

- One is bipartite and the other is not.
- They have distinct complements.
- They have different order or size.
- One graph has a vertex of degree k and the other doesn't.
- Their chromatic number is not the same.
- One contains a cycle and the other does not.

Showing that two graphs are *not* isomorphic

To show two graphs are not isomorphic, we can find a "structural property" (preserved by isomorphisms) on which they differ.

For example:

- One is bipartite and the other is not.
- They have distinct complements.
- They have different order or size.
- One graph has a vertex of degree k and the other doesn't.
- Their chromatic number is not the same.
- One contains a cycle and the other does not.
- They have different girth.

Showing that two graphs are *not* isomorphic

To show two graphs are not isomorphic, we can find a "structural property" (preserved by isomorphisms) on which they differ.

For example:

- One is bipartite and the other is not.
- They have distinct complements.
- They have different order or size.
- One graph has a vertex of degree k and the other doesn't.
- Their chromatic number is not the same.
- One contains a cycle and the other does not.
- They have different girth.

Definition

The **girth** of a graph is the length of its shortest cycle.