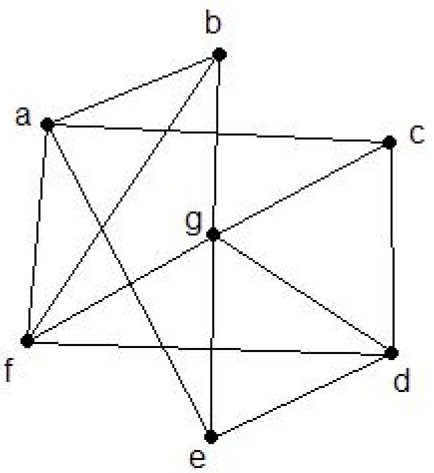
1 Basic Terminology

Definition 1.1. A *graph* is a set V of objects, called **vertices**, together with a set E of unordered pairs of distinct vertices in V, called **edges**. The edge corresponding to the unordered pair $\{u, v\}$ is denoted uv.

It is often useful to consider a visual interpretation of a graph. We represent the vertices by points on the plane, and edges as lines connecting the points representing the two vertices in the unordered pair.

Example 1.2. Let $V = \{a, b, c, d, e, f, g\}$, and $E = \{ab, ac, ae, af, bg, bf, cd, cg, de, df, dg, eg, fg\}$. Then $G = (V, E)$ is a graph. A graphical representation of G is given by



Definition 1.3. Let $G = (V, E)$ be a graph.
1. We say that vertices $u \in V$ and $v \in V$ are **adjacent** if $uv \in E$.
2. Given an edge $e = uv \in E$, we say that e is **incident** to u and v.
3. Given a vertex $v \in V$, we say the **neighbours** of v are $\{u \in V : u \text{ is adjacent to } v\}$.
4. Given a vertex $v \in V$, we say the **degree** of v is $|\{u \in V : u \text{ is adjacent to } v\}|$.

Example 1.4. In our graph G above, a is adjacent to b, c, e and f, so b, c, e and f are the neighbours of a. a has degree 4. The vertex c has degree 3, and has three incident edges: ac, cd, and cg.

2 Walks and Paths

Definition 2.1. A **walk** in a graph $G = (V, E)$ is an alternating sequence of vertices and edges $v_0, e_1, v_1, e_2, ..., v_{n-1}, e_n, v_n$ where each $e_i = v_{i-1}v_i$ is an edge in the graph for all $i = 1, 2, ..., n$. The **length** of a walk is the number of edges in the walk (in this case, n). We often denote a walk by its edges only (which uniquely determine the vertices in the walk), as $v_0v_1, v_1v_2, ..., v_{n-1}v_n$.

Definition 2.2. A **path** in a graph is a walk in which all vertices are distinct.

Definition 2.3. A **cycle** in a graph is a walk $v_0v_1, v_1v_2, ..., v_{n-1}v_0$ in which every v_i is distinct.

Theorem 2.4. *Let $G = (V, E)$ be a graph, and u,v be vertices in G. If there is a walk between u and v, then there is a path between u and v.*

Example 2.5. In our graph G above, a walk of length 5 between b and f is given by bg, gd, dc, cg, gf. A path of length 3 between a and f is given by af, fd, de. A cycle of length 4 in the graph G is given by ge, ed, dc, cg.

3 Subgraphs and Connectivity

Definition 3.1. Let $G = (V, E)$ be a graph. A **subgraph** of G is a set of vertices $V' \subseteq V$ together with a set of edges $E' \subseteq E$ where each edge has vertices only from V'.

Definition 3.2. We say that a graph G is **connected** if, for any vertices u and v in G, there is a path between u and v. A **connected component** of G is a maximal (by inclusion) connected subgraph of G.

Definition 3.3. Let $G = (V, E)$ be a connected graph. We say that an edge e is a **bridge** if $G - e = (V, E\backslash\{e\})$ is not connected.

Definition 3.4. We say a graph G is a **tree** if it is connected and does not contain any cycles.

Definition 3.5. Let $G = (V, E)$ be a connected graph. We say a subgraph $G' = (V', E')$ of G is a **spanning tree** if $V' = V$ and G' is a tree.

Theorem 3.6. *Every connected graph has a spanning tree.*

Example 3.7. In our graph G above, $G' = (V', E')$ where $V' = \{a, b, d, f, g\}$ and $E' = \{ab, bg, gf, gd\}$ is a connected subgraph of G. It is in fact a tree. The edge gb is a bridge of G'. The graph $G'' = (V, E'')$ where $E'' = \{ab, bg, gf, gd, de, gc\}$ is a spanning tree of G.

4 Bipartitions and Weightings

Definition 4.1. Let $G = (V, E)$ be a graph. A **cut** is a partition (A, B) of V into two sets, where $A \cup B = V$ and $A \cap B = \emptyset$. The **size** of the cut is the number of edges $uv \in E$ where $u \in S, v \in T$.

Definition 4.2. A graph $G = (V, E)$ is called **bipartite** if V can be partitioned into (A, B) where $A \cup B = V$ and $A \cap B = \emptyset$, andevery edge uv has $u \in A, v \in B$.

Definition 4.3. A **weighted graph** is a graph $G = (V, E)$ together with a function $w : E \to \mathbb{R}$ that maps edges to values called **edge weights** (or **edge costs**).

We will see in later lectures how the weight function is vital in formulating optimization problems.

5 Sample Theorems

Theorem 5.1. *In a tree T, for any vertices u and v, there is a unique path between u and v.*

Proof. If $u = v$, then the only path from u to v is the trivial path containing no edges, and we are done. So, we assume $u \neq v$.

Let $u = x_0 = y_0$. Suppose there are two distinct paths $p_1 = x_0x_1, x_1x_2, ..., x_kv$ and $p_2 = y_0y_1, y_1y_2, ..., y_lv$. Let n be the largest value such that $x_i = y_i$ for all $i = 0, 1, 2, ..., n$ (so then $x_{n+1} \neq y_{n+1}$). Such n exists since $p_1 \neq p_2$.

Let m_1 be the smallest value larger than n for which there is some m_2 such that $x_{m_1} = y_{m_2}$ (such a vertex exists since both paths arrive at v).

Then $x_nx_{n+1}, ..., x_{m_1-1}x_{m_1}$ and $y_ny_{n+1}, ..., y_{m_2-1}y_{m_2}$ are two paths from x_n to x_{m_1}, with no vertices in common. Thus, $C = x_nx_{n+1}, ..., x_{x_m-1}x_{m_1}, y_{m_2}y_{m_2-1}, y_{m_2-1}y_{m_2-2}, ..., y_{n+1}y_n$ is a cycle in T. This is a contradiction to the assumption that T is a tree and does not contain any cycles. Thus, there is a unique path from u to v in T. □

Theorem 5.2. *A graph $G = (V, E)$ is bipartite iff it has no cycles of odd length.*

Proof. ($\Rightarrow$) Let (A, B) be a bipartition of V. Suppose $C = x_1x_2, x_2x_3, ..., x_kx_1$ is a cycle of odd length (that is, k is odd). Assume WLOG that $x_1 \in A$. Then since $x_1x_2 \in E$, $x_2 \in B$. Since $x_2x_3 \in E, x_3 \in A$. Continuing this reasoning, then $x_i \in A$ if i is odd, and $x_i \in B$ is i is even. Since C is a cycle of odd length, k is odd. Thus, $x_k \in A$. But then the edge x_kx_1 has both endpoints in A, a contradiction. Thus, a bipartite graph cannot contain a cycle of odd length.

($\Leftarrow$) Assume G is connected, otherwise we can apply the following method to each of its connected components to get partitions $(A_1, B_1), ..., (A_n, B_n)$, and thus $(\cup_{i=1}^n A_i, \cup_{i=1}^n B_i)$ is a partition of G.

Choose any vertex r, and label it A. While there are still unlabelled vertices, find an unlabelled vertex v with a labelled neighbour. If the neighbour is labelled A, then label v with B, and if the neighbour is labelled B, then label v with A.

Suppose v has a neighbour u with the same label as v. Then the path from v to r through alternating labels intersects the path from u to r through alternating labels at some vertex w. Let p_1 be the path from v to w through alternating labels, and p_2 be the path from w to u through alternating labels. Since u and v share the same label, then the number of edges in p_1 and p_2 are both odd or both even. Thus, the concatenation of p_1 and p_2, together with the edge uv must be an odd cycle, a contradiction. Thus, no two adjacent vertices have the same label.

When there are no more unlabelled vertices, then (A, B) is bipartition of V, so G is bipartite.

□