Graphs

Definition: Graph

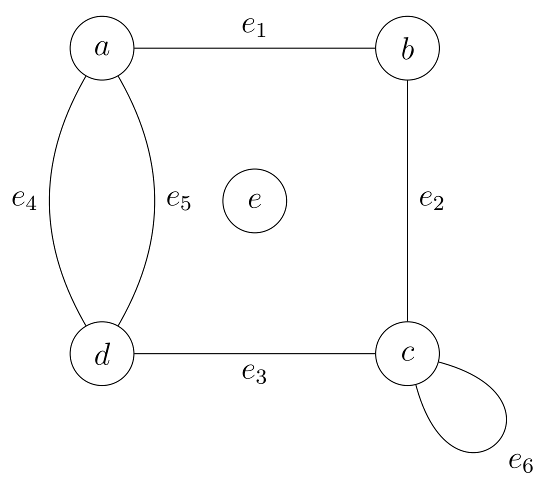
A *graph* is a mathematical object represented by a tuple $G = G(V, E, \ldots)$ consisting of a set of *vertices* (also called *nodes*) $V = V(G)$, a set of *edges* $E = E(G)$, and zero of more relations.

Each edge in $E(G)$ is associated with exactly two (not necessarily distinct) vertices in $V(G)$ called the *endpoints* of the edge. The nature of the edge/endpoint association is determined by the class of the graph (*undirected* vs *directed* and *multi* vs *simple*), which is indicated by the type of the elements in $E(G)$.

Each relation has $V(G)$ or $E(G)$ as its domain and is used to establish additional graph structure or to associate vertices or edges with problem-specific attributes (e.g. color, weight).

Example

Graphs are often portrayed visually using filled or labeled circles for the vertices and lines for the edges such that each edge line is drawn between its two endpoint vertex circles.



$V = V(G) = \{a, b, c, d, e\}$
$E = E(G) = \{e_1, e_2, e_3, e_4, e_5, e_6\}$

edge	endpoints
e_1	a, b
e_2	b, c
e_3	c, d
e_4	d, a
e_5	d, a
e_6	c, c

Note that it is not required that all vertices act as endpoints to edges; in the above example, vertex e is such a vertex.

Definition: Order

Let G be a graph. The *order* of G, typically denoted by $n = n(G)$, is the number of vertices in G:

$$n = n(G) = |V(G)|$$

Definition: Size

Let G be a graph. The *size* of G, typically denoted by $m = m(G)$, is the number of edges in G:

$$m = m(G) = |E(G)|$$

In the above example, $n = 5$ and $m = 6$.

Definition: Degenerate Cases

- The *null* graph is the graph with no vertices ($n = m = 0$).
- The *trivial* graph is the graph with exactly one vertex and no edges ($n = 1, m = 0$). Otherwise, the graph is *non-trivial*.
- An *empty* graph is a graph with no edges ($m = 0$).

Hence, both the null graph and the trivial graph are empty.

Definition: Labeled Graph

To say that a graph G is *labeled* means that its vertices are considered to be distinct and are assigned identifying names (labels) by adding a bijective labeling function to the graph tuple:

$$\ell : V(G) \to L$$

where L is a set of labels (names). Otherwise, the vertices are considered to be identical (only the structure of the graph matters) and the graph is *unlabeled*.

Since the labeling function ℓ is bijective, a vertex $v \in V(G)$ with label "a" can be identified by v or $\ell^{-1}(a)$. In practice, the presence of a labeling function is assumed for a labeled graph and so a vertex is freely identified by its label. This is important to note when a proof includes a phrase such as, "let $v \in V(G) \ldots$" since v may be a reference to any vertex in $V(G)$ or may call out a specific vertex by its label. The intention is usually clear from the context.