Chapter 8

Graph

8.1 Definition and terms

We commonly use the word *graph* to refer to a wide range of graphics and charts which provide an illustration or visual representation of information, particularly quantitative information. In the realm of mathematics, you probably most closely associate graphs with illustrations of functions in the real-number plane. *Graph theory*, our topic in this part, is a field of mathematics that studies a very specific yet abstract notion of a graph. It has many applications throughout mathematics, computer science, and other fields, particularly for modeling systems and representing knowledge.

Unfortunately, the beginning student of graph theory will likely feel intimidated by the horde of terminology required; the slight differences of terms among sources and textbooks aggravates the situation. The student is encouraged to take careful stock of the definitions in these chapters, but also to enjoy the beauty of graphs and their uses. This chapter will consider the basic vocabulary of graph theory and a few results and applications. The following chapter will explore various kinds of paths through graphs. Finally, we will use graph theory as a framework for discussing isomorphism, a central concept throughout mathematics.

A *graph* $G = (V, E)$ is a pair of finite sets, a set V of *vertices* (singular *vertex*) and a set E of pairs of vertices called *edges*. We will typically write $V = \{v_1, v_2, \ldots, v_n\}$ and $E = \{e_1, e_2, \ldots, e_m\}$ where each $e_k = (v_i, v_j)$ for some v_i, v_j; in that case, v_i and v_j are called *end points* of the edge e_k. Graphs are drawn so that vertices are dots and edges are line segments or curves connecting two dots.

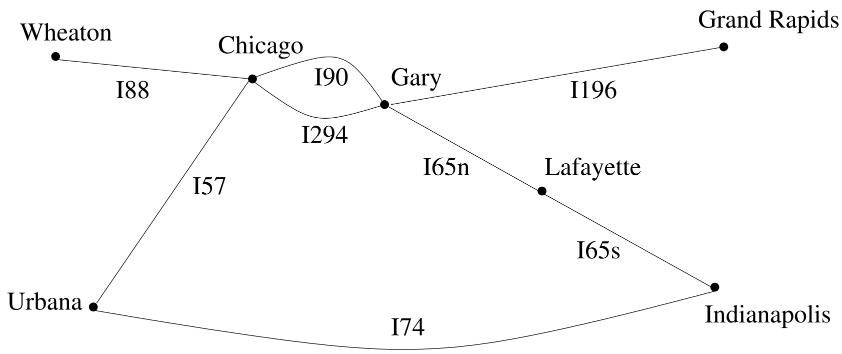
graph

vertex

edge

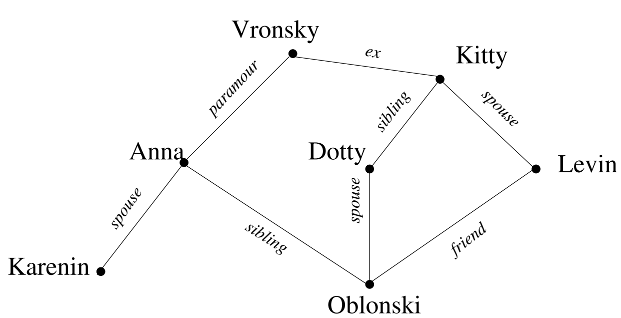
end points

As an example of a mathematical graph and its relation to everyday visual displays, consider the graph where the set of vertices is { Chicago, Gary, Grand Rapids, Indianapolis, Lafayette, Urbana, Wheaton } and the edges are direct highway connections between these cities. We have the following graph (with vertices and edges labeled). Notice how this resembles a map, simply more abstract (for example, it contains no accurate information about distance or direction).



We call the edges *pairs* of vertices for lack of a better term; a pair is generally considered a two-tuple (in this case, it would be an element of $V \times V$); moreover, we write edges with parentheses and a comma, just as we would with tuples. However, we mean something slightly different. First, tuples are ordered. In our basic definition of graphs, we assume that the end points of an edge are unordered: we could write I57 as (Chicago, Urbana) or (Urbana, Chicago). Second, an edge as a pair of vertices is not unique. In the cities example, we have duplicate entries for (Chicago, Gary): both I90 and I294. This is why it is necessary to have two ways to represent an edge, a unique name as well as a descriptive one.

The kinship between graphs and relations should be readily apparent. Graphs, however, are more flexible. As a second example, this graph represents the relationships (close friendship, siblinghood, or romantic involvement—labeled on the drawing but not as names for the edges) among the main characters of *Anna Karenina*. V = { Karenin, Anna, Vronsky, Oblonsky, Dolly, Kitty, Levin }. E = { (Karenin, Anna), (Anna, Vronsky), (Vronsky, Kitty), (Anna, Oblonsky), (Oblonsky, Dolly), (Dolly, Kitty), (Oblonsky, Levin), (Kitty, Levin) }.



incident

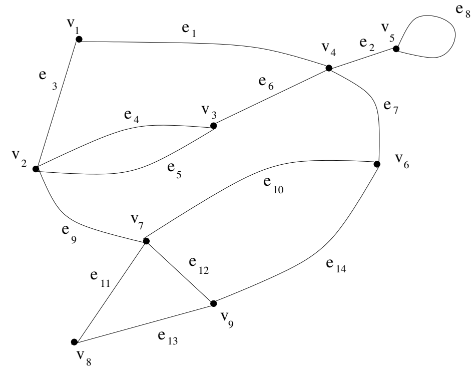
connects

adjacent

An edge (v_i, v_j) is *incident* on its end points v_i and v_j; we also say that it *connects* them. If vertices v_i and v_j are connected by an edge, they are *adjacent* to one another. If a vertex is adjacent to itself, that connecting edge is called a *self-loop*. If two edges connect the same two vertices, then those edges are *parallel* to each other. Below, e_1 is incident on v_1 and v_4. e_{10} connects v_7 and v_6. v_9 and v_6 are adjacent. e_8 is a self-loop. e_4 and e_5 are parallel.

self-loop

parallel

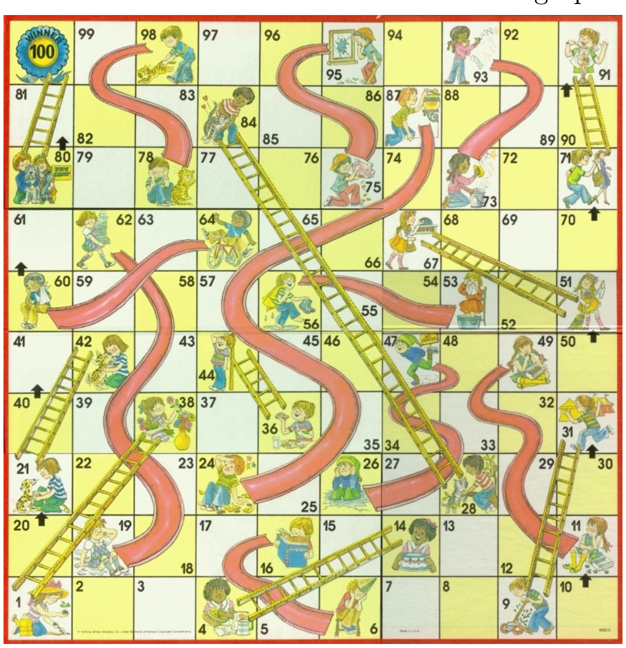


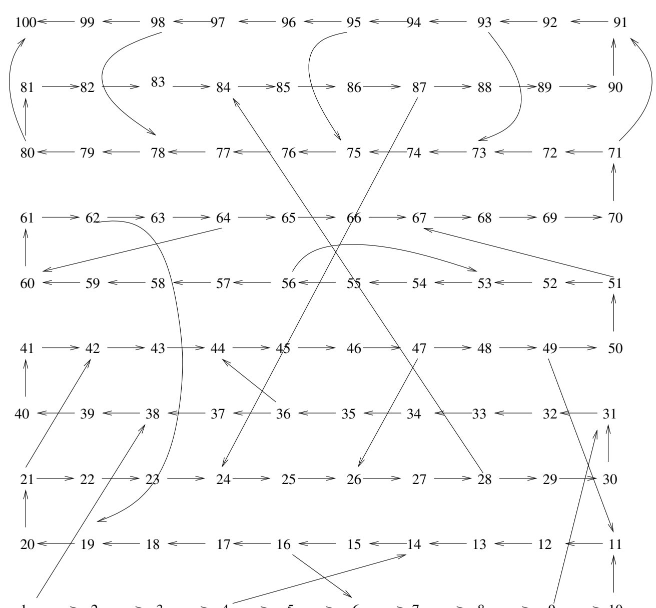
As we observed earlier, if a graph has parallel edges like e_4 and e_5 then the description (v_2, v_3) is not unique—it could describe either e_4 or e_5. If a graph has no parallel edges then a description like (v_2, v_3) can be used as the name of an edge.

The graphs described so far are called *undirected graphs* because the edges indicate a symmetric relationship between the two endpoints: if there is an edge $e = (v_1, v_2)$, then v_1 is adjacent to v_2 and v_2 is adjacent to v_1; we could have described this edge as (v_2, v_1). In a *directed graph*, sometimes abbreviated to *digraph*, the edges have a direction and are displayed with arrowheads. Consider the game board for Chutes and Ladders as a directed graph.

undirected graphs

directed graph





In Chapter 4 we used directed graphs to illustrate relations. In fact, a directed graph that has no parallel edges with the same direction is, formally, no different from a set together with a relation on that set (the original set being the vertices, and relation being the set of edges). In this chapter we will be more interested in undirected graphs. (Notice that undirected graphs that have no parallel edges are essentially the same thing as symmetric relations.)

degree

in-degree

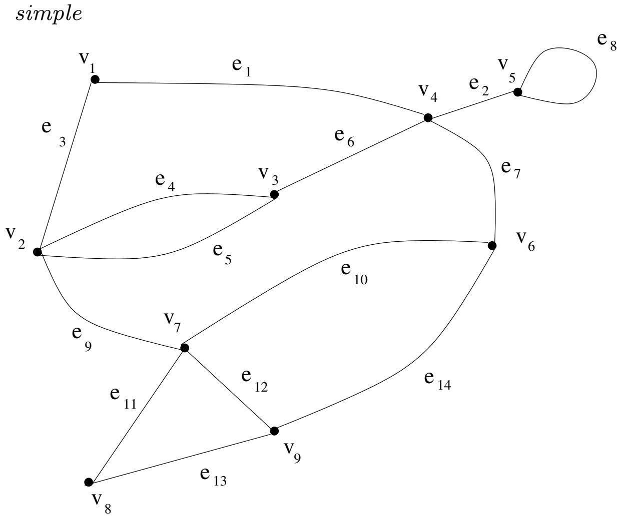
out-degree

The *degree* $\deg(v)$ of a vertex v is the number of edges incident on the vertex, with self-loops counted twice. $\deg(v_1) = 2$, $\deg(v_5) = 3$, and $\deg(v_2) = 4$. (In a directed graph, we would need to distinguish between the *in-degree* and *out-degree* of a vertex.)

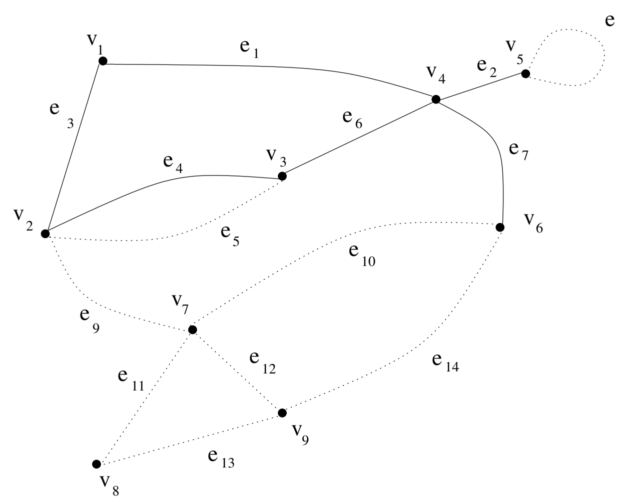
subgraph

simple

A *subgraph* of a graph $G = (V, E)$ is a graph $G' = (V', E')$ where $V' \subseteq V$ and $E' \subseteq E$ (and, by definition of graph, for any edge $(v_i, v_j) \in E'$, $v_i, v_j \in V'$). A graph $G = (V, E)$ is *simple* if it contains no parallel edges or self-loops.



sample graph

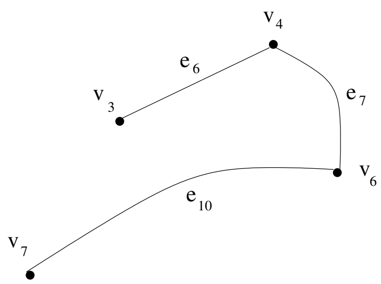


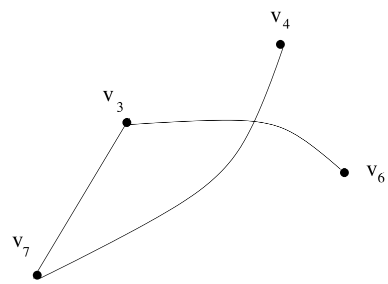
subgraph that happens to be simple

complete

complement

The graph $(\{v_1, v_2, v_3, v_4, v_5\}, \{e_1, e_2, e_3, e_4, e_6\})$ is a simple subgraph of the graph shown. A simple graph $G = (V, E)$ is *complete* if for all $v_i, v_j \in V$, the edge $(v_i, v_j) \in E$. The subgraph $(\{v_7, v_8, v_9\}, \{e_{11}, v_{12}, v_{13}\})$ is complete. The *complement* of a simple graph $G = (V, E)$ is a graph $\overline{G} = (V, E')$ where for $v_i, v_j \in V$, $(v_i, v_j) \in E'$ if $(v_i, v_j) \notin E$; in other words, the complement has all the same vertices and all (and only) those possible edges that are not in the original graph. The complement of the subgraph $(\{v_3, v_4, v_6, v_7\}, \{e_6, e_7, e_{10}\})$ is $(\{v_3, v_4, v_6, v_7\}, \{(v_3, v_7), (v_7, v_4), (v_3, v_6)\}$, as shown below.





Even though we tend to think of a graph fundamentally as a picture, it is important to notice that a formal description of a graph is independent of the way it is drawn. Important to the essence of a graph is merely the names of vertices and edges and their connections (and in Section **??**, we will see that even the names are not that important). The following two pictures show the same graph.