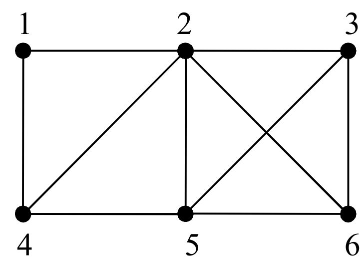
CMPS 101
Algorithms and Abstract Data Types

Graphs, Digraphs, and Trees

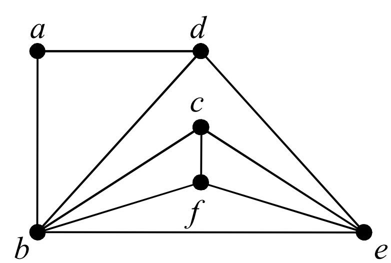
A *graph* G consists of an ordered pair of sets $G = (V, E)$ where $V \neq \varnothing$, and $E \subseteq V^{(2)} = \{2\text{-subsets of } V\}$, i.e. E consists of *unordered* pairs of elements of V. We call $V = V(G)$ the *vertex* set, and $E = E(G)$ the *edge* set of G. In this handout we consider only graphs in which both the vertex set and edge set are finite. An edge $\{x, y\}$, denoted *xy* or *yx*, is said to *join* its two *end vertices* x and y, and these ends are said to be *incident* with the edge *xy*. Two vertices are called *adjacent* if they are joined by an edge, and two edges are said to be *adjacent* if they have a common end vertex. A graph will usually be depicted as a collection of points in the plane (vertices), together with line segments (edges) joining the points.

Example $V(G) = \{1, 2, 3, 4, 5, 6\}$, $E(G) = \{12, 14, 23, 24, 25, 26, 35, 36, 45, 56\}$



Two graphs G_1 and G_2 are said to me *isomorphic* if there exists a bijection $\phi : V(G_1) \rightarrow V(G_2)$ such that for any $x, y \in V(G_1)$, the pair *xy* is an edge of G_1 if and only if the pair $\phi(x)\phi(y)$ is an edge of G_2. In other words, ϕ must preserve all incidence relations amongst the vertices and edges in G_1. We write $G_1 \cong G_2$ to mean that G_1 and G_2 are isomorphic.

Example Let G_1 be the graph from the previous example, and define G_2 by $V(G_2) = \{a, b, c, d, e, f\}$, $E(G_2) = \{ab, ad, bc, bd, be, bf, ce, cf, de, ef\}$. Define a map $\phi : V(G_1) \rightarrow V(G_2)$ by $1 \rightarrow a, 2 \rightarrow b, 3 \rightarrow c$, $4 \rightarrow d$, $5 \rightarrow e$, $6 \rightarrow f$. Clearly ϕ is an isomorphism. G_2 can be drawn as



Isomorphic graphs are indistinguishable as far as graph theory is concerned. In fact, graph theory can be defined to be the study of those properties of graphs that are preserved by isomorphism. Thus a graph is not a picture, in spite of the way we visualize it. A graph is a combinatorial object consisting of two abstract sets, together with some incidence data relating those sets.

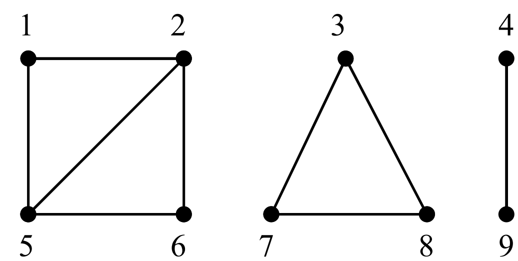
Given $x, y \in V(G)$ (not necessarily adjacent), a *walk* from x to y, or an x-y walk, is a sequence of vertices $x = v_0, v_1, v_2, \ldots, v_{k-1}, v_k = y$ such that $v_{i-1}v_i \in E(G)$ for $1 \le i \le k$. We call x the *origin* and y the *terminus* of the walk. These need not be distinct. If $x = y$, the walk is said to be *closed*. The *length* of the walk is *k*, the number of edge traversals performed in going from x to y along the sequence. Since the edges of a graph have no inherent direction, we do not distinguish between the above sequence and its reversal: $y = v_k, v_{k-1}, \ldots, v_2, v_1, v_0 = x$. Thus the designation as to which vertex in a walk is the origin and which is the terminus is arbitrary. A walk in which no edge is traversed more than once is called a *trail*, and a trail in which no vertex is visited more than once (except possibly when origin = terminus) is called a *path*. A closed path is called a *cycle*.

Example Refering to the above example we have:

a cycle of length 3: 2 5 6 2
a cycle of length 6: 1 2 3 6 5 4 1
a 1-6 path of length 5: 1 4 2 5 3 6
a 1-6 path of length 2: 1 2 6
a 3-1 trail which is not a path: 3 2 5 6 2 1
a 3-1 walk which is not a trail: 3 5 2 4 5 2 1

A graph G is said to be *connected* if it contains an x-y path for every $x, y \in V(G)$, otherwise G is called *disconnected*. The example above is clearly connected, while the following example is disconnected.

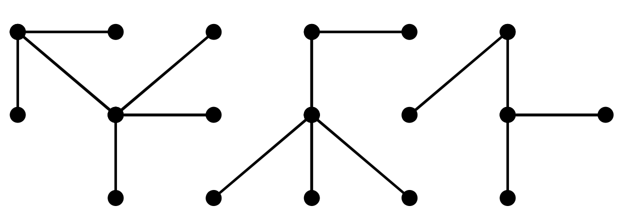
Example $V = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$ $E = \{12, 15, 25, 26, 56, 37, 38, 78, 49\}$



A *subgraph* of a graph G is a graph H in which $V(H) \subseteq V(G)$, and $E(H) \subseteq E(G)$. In the above example ({1, 2, 5}, {12, 15, 25}) is a connected subgraph, while ({2, 3, 6, 7}, {26, 37}) is a disconnected subgraph. A subgraph H is called a *connected component* of G if it is (i) connected, and (ii) maximal with respect to property (i), i.e. any other subgraph of G that contains H is disconnected. The above example clearly has three connected components: ({1, 2, 5, 6}, {12, 15, 25, 26, 56}), ({3, 7, 8}, {37, 38, 78}), and ({4, 9}, {49}). Obviously a graph is connected if and only if it has exactly one connected component.

A graph is called *acyclic* if it contains no cycles. A *tree* is a graph that is both connected and acyclic. The connected components of an acyclic graph are obviously trees. For this reason an acyclic graph is sometimes also called a *forest*. The following example is a forest with three connected components.

Example



Observe that in each tree of this forest, the number of edges is one less that the number of vertices. This fact holds in general for all trees. The following theorems demonstrate how the independent properties of connectedness and acyclicity are related.

Theorem 1 If T is a tree with n vertices and m edges, then $m = n - 1$.
Proof: See the induction handout, or the solutions to Midterm 1.

Theorem 2 If G is a connected graph with n vertices and m edges, then $m \geq n - 1$.
Proof: See the solutions to homework assignment 4.

Theorem 3 If G is a graph with n vertices, m edges, and k connected components, then $m \geq n - k$.
Proof:
Let $H_1, H_2, \ldots, H_k$, be the connected components of G. Let n_i and m_i denote the number of vertices and edges, respectively, of H_i, for $1 \leq i \leq k$. By Theorem 2 we have $m_i \geq n_i - 1$, for $1 \leq i \leq k$, and therefore

$$m = \sum_{i=1}^{k} m_i \geq \sum_{i=1}^{k} (n_i - 1) = \sum_{i=1}^{k} n_i - \sum_{i=1}^{k} 1 = n - k. \qquad ///$$

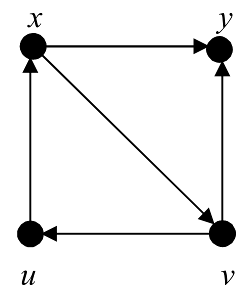
Theorem 4 If G is a forest with n vertices, m edges, and k connected components, then $m = n - k$.
Proof: Exercise. Hint: use Theorem 1, and emulate the proof of Theorem 3.

Theorem 5 Let G be a connected graph with n vertices and m edges. Suppose also that $m = n - 1$. Then G is acyclic, and hence a tree.
Proof:
Suppose G is connected and $m = n - 1$. Assume, to get a contradiction, that G is not acyclic. Let e be any edge belonging to any cycle in G. Remove e from G, and denote the resultant graph by $G - e$. Then $|E(G-e)| = m - 1 < m = n - 1$. However, since e is a cycle edge, its removal does not disconnect G, so $G - e$ is also connected. Theorem 2 above, applied to $G - e$, gives $|E(G-e)| \geq |V(G-e)| - 1 = n - 1$, and hence $|E(G-e)| < n - 1 \leq |E(G-e)|$, which is absurd. This contradiction shows that our assumption was false, and therefore G is acyclic. ///

Theorem 6 Let G be a forest with n vertices and m edges. Suppose also that $m = n - 1$. Then G is connected, and hence a tree.
Proof:
Suppose G is acyclic and $m = n - 1$. Let k be the number of connected components belonging to G. By Theorem 4 we have $m = n - k$, whence $n - 1 = n - k$, and therefore $k = 1$, showing that G is connected.
///

A *Directed Graph* (or *Digraph*) G is a pair of sets $G=(V,E)$, where V is finite and non-empty, and E consists of *ordered* pairs of elements of V, that is $E \subseteq V \times V$. We call V the vertex set and E the edge set of G. If there is more than one digraph under discussion we may write $V(G)$ and $E(G)$ to stand for these sets. Vertices are visually represented by points in the plane and directed edges as directed line segments.

Example 1 $V=\{x, y, u, v\}$ and $E=\{(x,y),(u,x),(v,y),(v,u),(x,v)\}$



We introduce some of the nomenclature surrounding directed graphs. The directed edge (x, y) in the above example is said to have *origin* x and *terminus* y, and we say that x is *adjacent* to y. The origin and terminus of a directed edge are said to be *incident* with that edge. Two edges are *adjacent* if they have a common end vertex, so for instance (x, y) is adjacent to (u, x). The *in degree* of a vertex is the number of edges having that vertex as terminus, and it's *out degree* is the number of edges having that vertex as origin. The *degree* of a vertex is the sum if it's in degree and out degree. Thus in the above example $\text{id}(x)=1$, $\text{od}(x)=2$, and $\deg(x)=3$.

A *directed path* P in a digraph is a finite sequence of vertices P: $v_0, v_1, v_2, \ldots, v_{k-1}, v_k$ such that $(v_{i-1}, v_i) \in E$ for all $1 \le i \le k$. The length of such a path is k, the number of edges traversed.