6c Lecture 3: April 7, 2015

3.1 Graphs and trees

We begin by recalling some basic definitions from graph theory.

Definition 3.1. A *(undirected, simple) graph* consists of a set of vertices V and a set $E \subseteq V \times V$ of *edges* with the property that for every $x, y \in V$ we have:

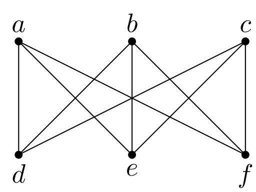
1. $(x, x) \notin E$.
2. $(x, y) \in E$ iff $(y, x) \in E$.

If $(x, y) \in E$, then we say that x and y are *adjacent*

Graphically, we represent graphs by drawing points to represent the vertices, and lines between them to represent the edges. For example, the graph whose vertices are $\{a, b, c, d, e, f\}$, and whose edges are

$$\{(a,d),(d,a),(a,e),(e,a),(a,f),(f,a),(b,d),(d,b),(b,e),(e,b),\\ (b,f),(f,b),(c,d),(d,c),(c,e),(e,c),(c,f),(f,c)\}$$

we can represent using the following picture:



Definition 3.2. A *path* from x to y in a graph is a finite sequence of distinct vertices $x_0, x_1, \ldots, x_n$ where $x_0 = x$, $x_n = y$, and for each $i < n$, (x_i, x_{i+1}) is an edge.

Definition 3.3. A graph is *connected* if for each two vertices x, y in the graph, there is a path from x to y.

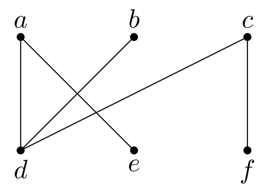
Definition 3.4. A *cycle* in a graph is a finite sequence of vertices $x_0, x_1, x_2, \ldots, x_n$, where $n \geq 3$, $x_0 = x_n$, (x_i, x_{i+1}) is an edge for every $i < n$, and $x_i \neq x_j$ for all $i, j < n$.

For example, the sequence a, d, b, f, a is a cycle in the above graph, while a, d, b, a is not (since (b, a) is not an edge), and neither is a, d, e, c (since a and c are not equal) or a, d, a (since the length must be at least 3).

Note that if $x_0, x_1, x_2, \ldots, x_n$ is a cycle, then the sequence $x_0, \ldots, x_{n-1}$ must be a path in the graph.

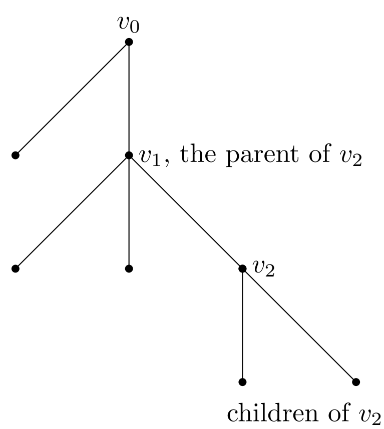
Definition 3.5. A *tree* is a connected graph having no cycles. Equivalently, a tree is a graph such that for every two vertices x, y in the graph, there is a unique path from x to y.

So for example, the graph we have drawn above is not a tree. However, the following graph is:



Definition 3.6. A *rooted tree* is a tree with a distinguished vertex called the root.

We will generally use v_0 to denote the root of a tree. Note that in a rooted tree, for every vertex v, there is a unique path $v_0, v_1, \ldots, v_n = v$ from the root to v. We say that a vertex is on *level* n when the length of this path is n. Give a vertex $v \neq v_0$ in a rooted tree, its *parent* is the vertex v_{n-1} such that $v_0, \ldots, v_{n-1}, v_n = v$ is the unique path from v_0 to v. The *children* of a vertex v is the set of all vertices having v as their parent.



3.2 König's lemma and compactness

Definition 3.7. An infinite branch of a rooted tree T is an infinite sequence starting at the root $v_0, v_1, v_2, \ldots$ such that for every i, v_{i+1} is a child of v_i.

Definition 3.8. A rooted tree is said to be *finitely splitting* if each node has only finitely many children.

Theorem 3.9 (König's lemma)**.** *Suppose T is a finitely splitting tree with infinitely many vertices. Then T has an infinite branch.*

Recall the pigeon-hole principle for infinite sets. If $S_0 \cup S_1 \cup \ldots \cup S_n$ is infinite, then some S_i must be infinite. (This is the contrapositive of the obvious fact that a finite union of finite sets is finite). The key to proving König's lemma is to repeatedly use the pigeon-hole principle to find a sequence $v_0, v_1, \ldots$ of vertices such that each v_i has infinitely many vertices below it in the tree.

Proof of König's lemma. Let v_0 be the root of T. Since T is finitely branching, we can list its children as $\{c_1, \ldots, c_k\}$ for some k. Define

$$S_i = \{v \in T \mid v \text{ is a descendent of } c_i\}.$$

Then since $T \setminus \{v_0\} = S_1 \cup S_2 \cup \ldots \cup S_k$, and $T \setminus \{v_0\}$ is infinite, the infinite pigeonhole principle tells us that there is some S_i which is infinite. Pick one, and define $v_1 = c_i$.

In general, suppose we have $v_0, v_1, \ldots, v_n$ a path in T, and v_n has infinitely many descendants. Since v_n has only finitely many children, the same reasoning as above shows that at least one of them has infinitely many descendants, and we let v_{n+1} be such a child. Since we can do this for any n, we get an infinite branch $v_0, v_1, v_2, \ldots$. □

Note that we need the condition that the graph is finitely splitting for König's lemma to be true. For example, consider the rooted tree whose root has infinitely many children labeled $\{1, 2, 3, \ldots\}$ but has no other vertices.

As a sidenote for those who know some topology, König's lemma is very closely related to compactness. For example, it is a good exercise to show that from König's lemma one can easily prove the Heine-Borel theorem that the unit interval is compact. That is, if $[0, 1]$ is covered by infinitely many open intervals of the form (a_i, b_i), then there is a finite subset of these open intervals which still covers $[0, 1]$. König's lemma also is easily seen to be a simple reformulation of the compactness of Cantor space.

We are now ready to give some applications of König's lemma. As an abstract principle, König's lemma excels at taking a collection of finite objects, and converting them into a single coherent infinite object.

3.3 The compactness theorem for propositional logic

We now use König's lemma to prove the compactness theorem for propositional logic. We will give two versions of the compactness theorem. The first is as follows:

Theorem 3.10 (The compactness theorem for propositional logic, I)**.** *If $S = \{\phi_1, \phi_2, \ldots\}$ is a set of formulas in the propositional variables $\{p_1, p_2, \ldots\}$, then S is satisfiable iff every finite subset of S is satisfiable.*

Proof. The proof was given in class, and was quite similar in spirit to Theorem **??**. The direction $\rightarrow$ is trivial. For the direction $\leftarrow$, we made a tree whose nth level consists of valuations of the variables $\{p_1, \ldots, p_n\}$ that do not make the formulas $\phi_1, \ldots, \phi_n$ false, arranged by compatibility. Then we showed that an infinite branch gave a valuation of $\{p_1, p_2, \ldots\}$ making all the formulas of S true. $\square$

We next give another version of the compactness theorem. However, first we will need the following lemma:

Lemma 3.11. *Suppose ϕ is a formula and S is a set of formulas. Then S implies ϕ iff $S \cup \{\neg\phi\}$ is unsatisfiable.*

Proof. If S implies ϕ, then every valuation making the formulas of S true make ϕ true. Hence, there is no valuation making all the formulas of S true and ϕ false. Hence, $S \cup \{\neg\phi\}$ is unsatisfiable.

Conversely, if $S \cup \{\neg\phi\}$ is unsatisfiable, it must be that every valuation making all the formulas of S true makes $\neg\phi$ false. Hence, every valuation making all the formulas of S true must make ϕ true. Hence S implies ϕ. $\square$

Theorem 3.12 (The compactness theorem for propositional logic, II)**.** *Suppose ϕ is a formula and $S = \{\phi_0, \phi_1, \ldots\}$ is a set of formulas. Then S implies ψ iff there is a finite subset $S' \subseteq S$ such that S' implies ϕ.*

Proof. S implies ψ iff $S \cup \{\neg\psi\}$ is unsatisfiable (by Lemma 3.11) iff there is a finite subset of $S \cup \{\neg\psi\}$ that is unsatisfiable (by the compactness theorem) iff there is a finite subset $S' \subseteq S$ such that $S' \cup \{\neg\psi\}$ is unsatisfiable iff there is a finite subset $S' \subseteq S$ such that S' implies ψ (by Lemma 3.11). $\square$

It is a good exercise to show that version II of the compactness theorem also easily implies version I.