Chapter 5. Counting

Section 5.2. Cayley's Spanning Tree Formula

Note. Recall that a spanning tree of a graph G is a subtree of G that contains all vertices of G. In Graph Theory 1 (MATH 5340), it is shown that the number of *labeled* trees on n vertices is n^{n-1} and it is argued that the number of labeled treed on n vertices is equal to the number of spanning trees of K_n (see my online notes for that class on Section 4.2. Spanning Trees; notice Theorem 4.8). While considering the number of hydrocarbons of a certain type (those without "cycles"), Arthur Cayley (1821-1895) represented atoms as vertices and chemical bonds as edges. In "A Theorem on Trees," *Quarterly Journal of Pure and Applied Mathematics*, **23** (1889), 376–378, he proved that the number of spanning trees on n (labeled) vertices is n^{n-2}. This is now called "Cayley's Formula." In this section we give a proof using sequence of elements of $\{1, 2, \ldots, n\}$ associated with a spanning tree of K_n (these sequences are called "Prüfer codes"). This approach is to be used in Exercise 4.2.11 in J. A. Bondy and U. S. R. Murty's *Graph Theory*, Graduate Texts in Mathematics #244 (Springer, 2008); this is the text used in the graduate Graph Theory sequence (MATH 5340, MATH 5450).

Theorem 5.2.1. Cayley's Formula. The number of spanning trees in K_n is $s(K_n) = n^{n-2}$.

Note. By considering spanning trees in a given K_n, we are effectively considering the vertices as labeled. The effect of this is that we distinguish between subtrees that are isomorphic. We are not counting the number of tree on n vertices "up to isomorphism"; this problem is unsolved according to Hartsfield and Ringel (see page 95).

Note. Cayley's Formula "clearly" holds for $n = 3$, since a complete graph on three vertices yield $n^{n-2} = (3)^{(3)-2} = 3$ trees (each determined by deleting a single edge of K_3). For $n = 4$, there are $n^{n-2} = (4)^{(4)-2} = 16$ spanning trees of K_4, as given in Figure 5.2.1 (notice that the label of a vertex is implied by location; in fact, there are only two trees on 4 vertices "up to isomorphism," namely $K_{1,3}$ and P_3). For the general proof, we use the following lemma.

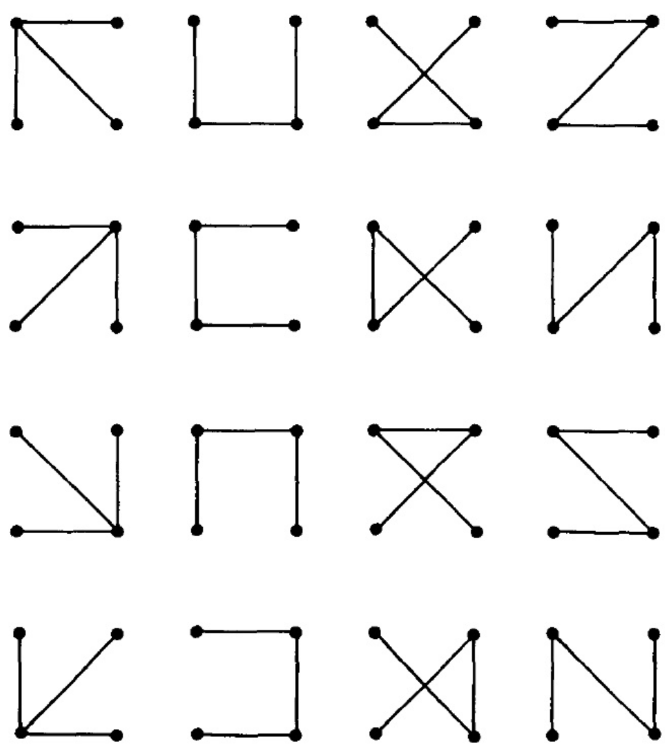
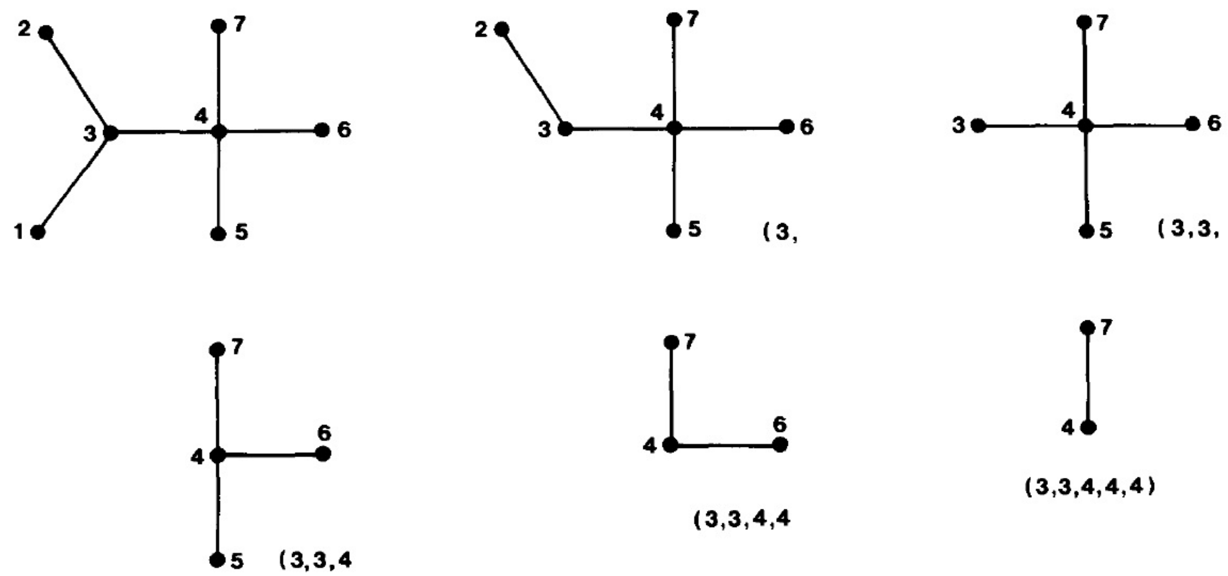


Figure 5.2.1

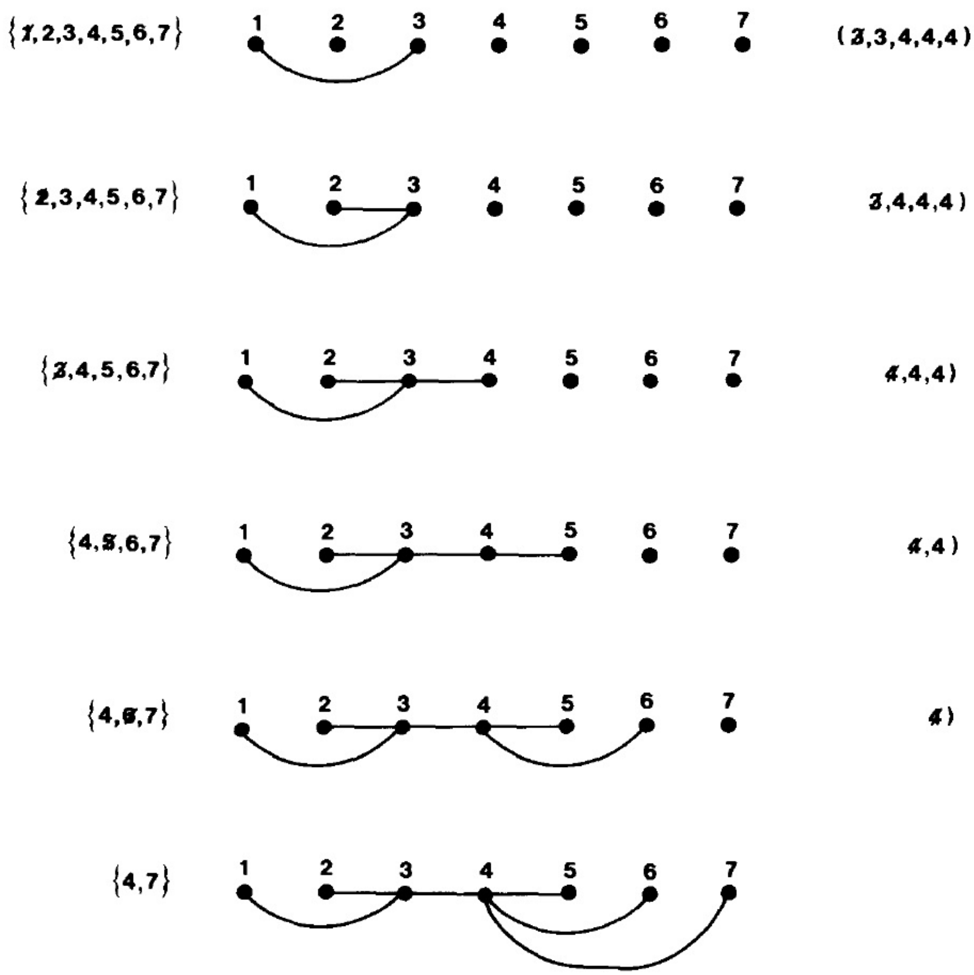
Lemma 5.2.A. The number of different sequences $(b_1, b_2, \ldots, b_{n-2})$ of length $n-2$, where $b_i \in \{1, 2, \ldots, n\}$ and repetition is allowed, is n^{n-2}.

Note. To prove Cayley's Formula (Theorem 5.2.1), we establish a one-to-one correspondence (i.e., a bijection) between the set of spanning trees of K_n and the set of sequences of length $n-2$ from $\{1, 2, \ldots, n\}$. We now illustrate the creation of the sequence (or Prüfer code). Consider the tree T with $n = 7$ vertices as follows:



We think of $N = \{1, 2, \ldots, 7\}$ as an ordering (and a labeling) of the vertices of T. We'll denote the associated sequence as $(t_1, t_2, t_3, t_4, t_5)$. First we consider the vertex of degree 1 with the smallest label; this is vertex 1. Next, we set t_1 equal to the vertex label of the one neighbor of vertex 1; this is $t_1 = 3$. Then we delete vertex 1 from T to create tree T_1 (above, first row center). We then repeat the process so that the smallest label of a vertex of degree 1 in T_1 is 2. We set t_2 equal to the vertex label of the one neighbor of vertex 2; that is $t_2 = 3$. Delete vertex 2 from tree T_1 to create tree T_2 (above, first row right). The smallest label of a vertex of degree 1 in T_2 is 3. We set t_3 equal to the vertex label of the one neighbor of vertex 3; that is $t_3 = 4$. Delete vertex 3 from tree T_2 to create tree T_3 (above, second row left). The smallest label of a vertex of degree 1 in T_3 is vertex 5. We set t_4 equal to the vertex label of the one neighbor of vertex 5; that is $t_4 = 4$. Delete vertex 5 from tree T_3 to create tree T_4 (above, second row center). The smallest label of a vertex of degree 1 in T_4 is vertex 6. We set t_5 equal to the vertex label of the one neighbor of vertex 6; that is $t_5 = 4$. Delete vertex 6 from tree T_4 to create tree T_5 (above, second row right). We stop here. The Prüfer code is $(t_1, t_2, t_3, t_4, t_5) = (3, 3, 4, 4, 4)$. Notice that the degree of vertex 3 in original tree T is three and 3 appears twice in the Prüfer code. The degree of vertex 4 in T is four and 4 appears three times in the Prüfer code. In fact, if a vertex v has degree k in the original tree, then number v will appear $k - 1$ times in the sequence; therefore, the degree 1 vertices do not appear in the Prüfer code.

Note. We now illustrate the reversal of the above process. That is, we start with the sequence $(3, 3, 4, 4, 4)$ and construct tree T. We start with the smallest element of the set $N = \{1, 2, 3, 4, 5, 6, 7\}$ that does *not* appear in the sequence. This is 1. We connect 1 to the first vertex in the sequence, vertex 3. Then we delete vertex 1 from set N to get $N_1 = N \setminus \{1\}$ and delete the first entry in the sequence to get the new shortened sequence $(3, 4, 4, 4)$ (see below top). Now 2 is the smallest vertex label in $N_1 = \{2, 3, 4, 5, 6, 7\}$ not in sequence $(3, 4, 4, 4)$, so we connect it to the first vertex in sequence $(3, 4, 4, 4)$, vertex 3. Then we define $N_2 = N_1 \setminus \{2\}$ and delete the first entry of the shortened sequence to get $(4, 4, 4)$ (see below second from top). Now 3 is the smallest vertex label in $N_2 = \{3, 4, 5, 6, 7\}$ not in sequence $(4, 4, 4)$, so we connect it to the first vertex in sequence $(4, 4, 4)$. Then we define $N_3 = N_2 \setminus \{3\}$ and delete the first entry of the shortened sequence to get $(4, 4)$ (see below third). Now 5 is the smallest vertex label in $N_3 = \{4, 5, 6, 7\}$ not in sequence $(4, 4)$, so we connect it to the first vertex in sequence $(4, 4)$. Then we define $N_4 = N_3 \setminus \{5\}$ and delete the first entry of the shortened sequence to get (4) (see below fourth). Now 6 is the smallest vertex label in $N_4 = \{4, 6, 7\}$ not in sequence (4), so we connect it to the first vertex in sequence (4). Then we define $N_5 = N_4 \setminus \{6\}$ and delete the only entry of the shortened sequence (4) so that the sequence is gone (see below fifth). Finally, we connect the last two numbers in $N_5 = \{4, 7\}$ (see below last).



Notice that the tree above is the same as the tree T in the creation of the Prüfer code $(3, 3, 4, 4, 4)$ above. Hartsfield and Ringel use the above examples, along with an argument that the process of creating the Prüfer code from a tree and the process of creating a tree from a Prüfer code are (in general) inverses of each other. We prefer a cleaner proof and now present the proof of Cayley's Formula (Theorem 5.2.1) given in J.A. Bondy and U.S.R. Murty's *Graph Theory with Applications*, Macmillan Press, 1976 (see Theorem 2.9 in their Section 2.4 "Cayley's Formula"). This proof originally appeared in Heinz Prüfer's "Neuer Beweis eines Satzes über Permutationen [New Proof of a Theorem on Permutations]," *Archiv der Mathematischen Physik*, **27**(3), 142–144 (1918).

Revised: 12/8/2022