Math 38. Graph Theory.

Solutions to Homework 4.

2.1.15. The contrapositive of Thm 2.1.11 says that if $\text{diam}(\bar{G}) > 3$, then $\text{diam}(G) < 3$. So, if a simple graph has diameter ≥ 4, then its complement has diameter ≤ 2.

2.1.23. Let n be the total number of vertices, and let m be the number of vertices of degree k (so $n-m$ is the number of vertices of degree 1).

The sum of the degrees of all the vertices is then $mk + (n-m) = 2e(T) = 2n-2$, using the degree-sum formula and the fact that trees have $n-1$ edges. It follows that

$$n = 2 + (k-1)m.$$

Finally, we show that each such value of n, where m can be chosen to be any nonnegative integer, is possible. For $m = 0$, take the tree with 2 vertices and 1 edge. For $m > 0$, consider a path with vertices $v_1, \ldots, v_m$, where each of v_1 and v_m is adjacent to $k-1$ additional vertices, and each of $v_2, \ldots, v_{m-1}$ is adjacent to $k-2$ additional vertices. Such a tree has $n = 2 + (k-1)m$ vertices.

2.1.27. Forward direction: If there is a tree with vertex degrees $d_1, \ldots, d_n$, then, by the degree-sum formula,

$$\sum d_i = 2e(T) = 2n-2,$$

using the fact that trees have $n-1$ edges.

Backward direction: Use induction on n. For $n = 2$, the only way to have $d_1 + d_2 = 2$ is with $d_1 = d_2 = 1$. The tree with one edge has these degrees.

Let $n > 2$, and assume the statement is true for smaller values of n. The fact that $\sum d_i = 2n-2$ shows that $n < \sum d_i < 2n$. The first inequality implies that there is some $d_i > 1$, while the second inequality implies that there is some $d_j = 1$. Without loss of generality, assume that $d_1 > 1$ and $d_n = 1$. Let $d'_1 = d_1 - 1$, and $d'_k = d_k$ for $2 \leq k \leq n-1$. Then

$$\sum_{i=1}^{n-1} d'_i = 2(n-1) - 2,$$

and all the d'_i are positive, so we can apply the induction hypothesis to conclude that there is a tree with vertex degrees $d'_1, \ldots, d'_{n-1}$. Adding an edge between the vertex of degree d'_1 and a new vertex, we get a tree with degrees $d_1, \ldots, d_n$ as desired.

2.1.32. <u>e cut-edge $\Rightarrow$ e belongs to every spanning tree:</u> A spanning tree of G not containing e would be a spanning tree of $G - e$. But such a spanning tree cannot exist because $G - e$ is disconnected, since by assumption e is a cut-edge.

<u>e belongs to every spanning tree $\Rightarrow$ e cut-edge:</u> Let us prove the contrapositive. If e is not a cut-edge, then $G - e$ is connected, so it has a spanning tree by Corollary 2.1.5(c). Such a

spanning tree would also be a spanning tree of G, since it has the same vertex set, and it would not contain e.

e loop $\Rightarrow$ e belongs to no spanning tree: A loop is a cycle of length 1, but trees are acyclic, so e cannot be in any spanning tree.

e belongs to no spanning tree $\Rightarrow$ e loop: Let us prove the contrapositive. If e is not a loop, we can construct a spanning tree containing e as follows. Starting with the graph with vertex set $V(G)$ and edge set $\{e\}$, repeat this process: if the graph is disconnected, add an edge of G between two of the components. Such an edge always exists because G is connected, and adding it does not create a cycle. We repeat the process until the resulting graph is connected. We obtain a subgraph of G that is connected and acyclic (that is, a tree) with vertex set $V(G)$, so it is a spanning tree of G containing e.

2.2.1. In this problem we use the fact that the vertices not appearing in the Prüfer code are precisely the leaves of the tree.

(a) Only the star $K_{1,n-1}$, because such trees must have $n-1$ leaves.

(b) Trees with exactly $n-2$ leaves. These consist of an edge uv together with with i additional vertices adjacent to u and $n-i-2$ additional vertices adjacent to v, where $1 \leq i \leq n-3$.

(c) Such a tree must have 2 leaves, and all other vertices must have degree 2, so it must be a path.

2.2.7. Fix an edge of K_n, and let x be the number of spanning trees of K_n that contain that edge. Then the number we are trying to find, that is, the number of spanning trees of K_n that do not contain that edge, is $n^{n-2} - x$, since, by Cayley's formula, the total number of spanning trees of K_n is n^{n-2}.

Let M be the number of pairs (T, e) where T is a spanning tree of K_n and e is an edge of T. We find two formulas for M. On the one hand, $M = n^{n-2}(n-1)$, since we can first choose a tree T in n^{n-2} ways (by Cayley's formula) and then choose an edge of the tree in $n-1$ ways (because each tree has $n-1$ edges). On the other hand, $M = \binom{n}{2}x$, since we can first choose an edge of K_n in $\binom{n}{2}$ ways, and then choose a spanning tree containing that edge.

Equating the two formulas for M and solving for x, we get that

$$x = \frac{n^{n-2}(n-1)}{\binom{n}{2}} = 2n^{n-3},$$

and so the number of spannning trees that do not contain a given edge is $n^{n-2} - 2n^{n-3} = (n-2)n^{n-3}$.

2.2.8. a) Direct argument: These are paths, where the vertices have labels from the set $[n]$. We have $n!$ ways to assign labels to the vertices, but such assignments count each labeled path twice, since reversing the labels produces the same labeled path. Thus, the number of such trees is $n!/2$.

Using Prüfer code: The Prüfer code of such trees has $n-2$ different entries. We have n choices for the first entry, $n-1$ for the second..., and finally 3 for the last entry. Thus, the number of such trees is $n(n-1)(n-2)\cdots 3 = n!/2$.

b) Direct argument: These trees consist of an edge uv together with with i additional vertices adjacent to u and $n-i-2$ additional vertices adjacent to v, where $1 \leq i \leq n-3$. The pair $\{u, v\}$ can be chosen in $\binom{n}{2}$ ways. Then, the neighbors of u can be any subset of the remaining $n-2$ vertices, except for the empty subset and for the whole set. Thus, there are $2^{n-2}-2$ choices for the neighbors of u, and each such choice determines the neighbors of v as well. In total, there are $\binom{n}{2}(2^{n-2}-2) = n(n-1)(2^{n-3}-1)$ such trees.

Using Prüfer code: The Prüfer code for such trees contains exactly two values. We have $\binom{n}{2}$ ways to choose these two values, say $\{u, v\}$. After these values are chosen, we can choose in which among the $n-2$ positions of the Prüfer code to put u, and then put v in the remaining positions. The choice of positions for u can be any proper (i.e. other than empty or the whole set) subset of the $n-2$ positions, so there are again $2^{n-2}-2$ choices, giving a total of $\binom{n}{2}(2^{n-2}-2)$ trees.