Math 108 Homework 2 Solutions

Problem 2B
How many trees T are there on the set of vertices $\{1, 2, 3, 4, 5, 6, 7\}$ in which the vertices 2 and 3 have degree 3, vertex 5 has degree 2, and hence all others have degree 1? Do not just draw pictures but consider the possible Prüfer codes of these trees.

Proof. As noted in Proof 1 of Theorem 2.1 in the text, the number of times a vertex v occurs among $y_1, y_2, ..., y_{n-2}$ in the Prüfer code of a given tree T is equal to $\deg_T(v) - 1$, where $\deg_T(v)$ is the degree of v in T. Thus, a tree T on the vertices $\{1, 2, 3, 4, 5, 6, 7\}$ satisfies these degree requirements if and only if its Prüfer code has two 2s, two 3s, and one 5. The number of distinct Prüfer codes with these entries is given by $\frac{5!}{2!2!1!} = 30$. Because of the bijective correspondence between spanning trees and Prüfer codes, we conclude that **there are precisely 30 trees on the vertices $\{1, 2, 3, 4, 5, 6, 7\}$ in which vertices 2 and 3 have degree 3, vertex 5 has degree 2, and all others have degree 1.** $\square$

Problem 2D
Here is a variation on the above greedy algorithm. Let x_1 be any vertex of a weighted connected graph G with n vertices and let T_1 be the subgraph with the one vertex x_1 and no edges. After a tree (subgraph) $T_k, k < n$, has been defined, let e_k be a cheapest edge among all edges with one end in $V(T_k)$ and the other end not in $V(T_k)$, and let T_{k+1} be the tree obtained by adding that edge and its other end to T_k. Prove that T_n is a cheapest spanning tree in G.

Proof. This algorithm is known as Prim's algorithm, and proofs of its validity are widespread online.

Suppose that Prim's algorithm does not generate a cheapest spanning tree. Let $T_0 = \emptyset$, and let k be chosen minimally to satisfy the property that there is no cheapest spanning tree containing all of the edges in T_k (since $T_0 = \emptyset$, we know $k \geq 1$.) Let M be a cheapest spanning tree containing T_{k-1}. Note that $e_k \notin M$ since then $T_k \subseteq M$, but by hypothesis no cheapest spanning tree contains T_k.

Let M' be the graph obtained by adding the edge e_k to M. Then M' is a connected graph with n edges on n vertices, and M' contains a cycle. Note that removing any edge from that cycle results in a connected graph with $n-1$ edges on n vertices, and so results in a spanning tree by homework 1.

Since e_k is an edge which leaves T_{k-1} and is a member of a cycle in M', there must be another edge e^* in that cycle which also leaves T_{k-1}. By Prim's algorithm, the cost of

e^* is at least that of e_k. Thus, if we obtain the graph M'' by removing the edge e^* from M', we obtain a spanning tree which is of cost no higher than that of M. But this M'' is then a cheapest spanning tree which contains T_k, contradicting the definition of k.

Thus, we may conclude that for all $k \leq n$, there is a cheapest spanning tree containing T_k. In particular, there is a cheapest spanning tree containing T_n, but since T_n is already a spanning tree, T_n must be a cheapest spanning tree. □

Problem 2F

Suppose a tree G has exactly one vertex of degree i for $2 \leq i \leq m$ and all other vertices have degree 1. How many vertices does G have?

Proof. Recall that the number of edges in G is given by $\frac{1}{2}\sum_{v\in V}\deg_G(v)$. By hypothesis, G has $m-1$ vertices with degree greater than one, and so if G has n vertices, then G has $n-m+1$ vertices with degree 1. Thus, we have that the number of edges in G is equal to $\frac{1}{2}\left(n-m+1+\sum_{i=2}^{m} i\right) = \frac{1}{2}\left(n-m+\frac{m(m+1)}{2}\right) = \frac{2n+m(m-1)}{4}$.

Recall also from Homework 1 that if G is a tree with n vertices, then G has $n-1$ edges. Multiplying these two formulas for the number of edges by four and comparing them, we see that $4n-4 = 2n+m(m-1)$, so that $n = \frac{m(m-1)}{2}+2$. **Thus, the tree G has $\mathbf{\frac{m(m-1)}{2}+2}$ vertices.** □

Problem 3B

Let the edges of K_7 be colored with the colors red and blue. Show that there are at least four subgraphs K_3 with all three edges the same color (monochromatic triangles). Also show that equality can occur.

Proof. Define a biangle to be a subgraph of K_7 consisting of three nodes and two edges, and call the node adjacent to both edges the base of the biangle. Call the biangle non-monochromatic if the edges are different colors, otherwise call it monochromatic. Note that the edges in any biangle are a part of a unique triangle, and that any non-monochromatic triangle contains precisely two non-monochromatic biangles.

In K_7, every vertex has six edges to which it is adjacent. If the vertex v is adjacent to r red edges, then v is the base of $r(6-r) \leq 9$ non-monochromatic biangles. Thus, any coloring of the edges of K_7 contains at most $63 = 9 \cdot 7$ non-monochromatic biangles, since every non-monochromatic biangle has some vertex as its base, there are seven vertices in K_7, and each vertex in K_7 is the base of at most 9 non-monochromatic biangles.

Since every non-monochromatic triangle contains precisely two non-monochromatic biangles, the number of non-monochromatic triangles is equal to one half the number of non-monochromatic biangles. Therefore, in K_7, the number of non-monochromatic triangles is bounded above by $63/2 = 31.5$, and since it must be an integer, it is bounded

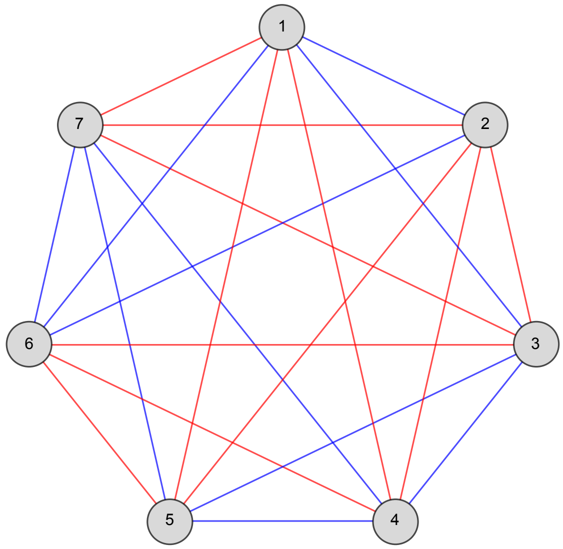

Figure 1: An edge-coloring of K_7 with precisely four monochromatic triangles.

above by 31. Since K_7 contains $\binom{7}{3} = 35$ triangles, we may conclude that at least $35-31 = 4$ of these must be monochromatic.

Also, because every non-monochromatic triangle contains precisely two non-monochromatic biangles, to show that there is an edge-coloring of K_7 with precisely four monochromatic triangles, it is equivalent to construct an edge-coloring of K_7 with precisely $(35-4)*2 = 62$ non-monochromatic biangles.

This is achieved in Figure 1, since 6 of the vertices (all except vertex 2) have three adjacent red edges and three adjacent blue edges (and so are the bases of a total of $6 * 9 = 54$ non-monochromatic biangles), and vertex 2 is adjacent to four red edges and two blue, for a total of $54+8 = 62$ non-monochromatic biangles. Thus, this edge-coloring must give rise to precisely four monochromatic triangles.

$\square$