Maggie Johnson
CS109 Handout #21a

Problem Set #6 Solutions

1) Such a tree does exist. By Theorem 4(iii) we note that such a tree has $i = 75/(m-1)$ internal vertices. This has to be a whole number so $(m-1)$ must divide 75. This is possible, for example, with $m = 6$. A complete 6-ary tree of height 2 would have 36 leaves. We need to add 40 leaves. This can be done by changing 8 vertices at level 2 to internal vertices; each such change adds 5 leaves to the tree (6 new leaves at level 3, less the one leaf at level 2 that has been changed to an internal vertex). The tree looks like this: root has 6 children each of which has 6 children giving 36 vertices at level 2. Of these, 28 are leaves and the remaining 8 vertices at level 2 has 6 children, giving us 48 leaves at level 3. The total number of leaves is 28 + 48 = 76 which is what we wanted.

2a) This tree has one vertex at level 0, m vertices at level 1, m^2 vertices at level 2, m^h vertices at level h. Therefore it has:

$$1 + m + m^2 + \ldots + m^h = (m^{h+1} - 1) / (m-1)$$ vertices in all. (This formula was proven in the Induction handout.)

At level h where the leaves are, we have m^h elements.

2b) (We assume $m >= 2$). First we delete all the vertices at level h; there is at least one such vertex and they are all leaves. The result must be an m-ary tree of height h-1. The result of the previous exercise tells us that this tree has m^{h-1} leaves. In the original tree, then, there are more than this many leaves since every internal vertex at level h-1 (which counts as a leaf in our reduced tree) spawns at least two leaves at level h.

** Note: if you just said “there will be more leaves at level h” is not sufficient. We wanted some sort of statement about how for every leave that is turned into an internal node at level h-1, it spawns at least two leaves at level h, hence adding to m^{h-1} because $(2 – 1) = 1$. Some people did a proof using the ceiling function shown below. This is acceptable as well.

$h = \text{ceil}(\log_m l)$	(Formula in the textbook)
$h-1 < \log_m l <= h$	definition of a ceiling function
$m^{h-1} < l <= m^h$	
$m^{h-1} < l$	

3) recurrence relation for # vertices: $v_1 = 1$, $v_2 = 1$; $v_n = v_{n-1} + v_{n-2} + 1$ (the "+1" is for the root) for $n>= 3$
closed form for # vertices: $2f_n - 1$ where f_n is the nth Fibonacci number.

P(n) denotes: a closed form formula for $v_1 = 1$, $v_2 = 1$; $v_n = v_{n-1} + v_{n-2} + 1$ is $2f_n - 1$ where f_n is the n^{th} Fibonacci number.
<u>base case:</u> $v_1 = (2 * 1) - 1 = 1$; $v_2 = (2 * 1) - 1 = 1$ (F(1) = 1, F(2) = 1)
<u>inductive hypothesis</u>: Assume that for all positive numbers i where $i <= k$: $v_i = 2f_i - 1$ and show P(k+1): $v_{k+1} = 2f_{k+1} - 1$

<u>PROOF:</u>

$v_{k+1} = v_k + v_{k-1} + 1$	from the recurrence relation
$v_{k+1} = (2f_k - 1) + (2f_{k-1} - 1) + 1$	substitute inductive hypothesis
$v_{k+1} = 2f_k + 2f_{k-1} - 1 + 1 - 1$	algebra

$v_{k+1} = 2f_k + 2f_{k-1} - 1$
$v_{k+1} = 2f_{k+1} - 1$ from the recurrence relation

P(k+1) is true when P(k) is true, and therefore P(n) is true for all integers >= 1.

The number of leaves satisfies the recurrence relation $l_n = l_{n-1} + l_{n-2}$ with $l_1 = l_2 = 1$, so $l_n = f_n$. Since $i_n = v_n - l_n$, we have $i_n = f_n - 1$. Finally, it is clear that the height of the tree T_n is one more than the height of the tree T_{n-1} for n >= 3 (with the height of T_2 being 0) since all we have added to the trees T_{n-1} and T_{n-2} is a root to get T_n. Thus, the height of T_n is n-2 for n >= 2.

4) Is it possible to have a group of seven people such that each person knows exactly three other people in the group? No: If we try and model this using a graph with a vertex for each person and an edge between pairs of people who know each other, then we would have a graph with 7 vertices all of odd degree. This is impossible by our theorem: The number of vertices of odd degree must be even.

5) Suppose all vertices of a graph G have degree p, where p is an odd number. Prove that the number of edges in G is a multiple of p.

We know that p * V = 2E by the Handshaking Theorem. Therefore, E = p * V/2. 2 cannot divide p because p is an odd number, so E = p * (V/2), and thus, E is a multiple of p.

6) Suppose the parts of the bipartite graph are of sizes k and v-k. Then, the maximum number of edges the graph may have is k(v-k) (an edge between each pair of vertices in different parts). Now we have to determine at what value of k the function f(k) = k(v-k) reaches its maximum. By algebra or calculus, we see it achieves a maximum at $f(k) = v^2/4$. Thus there are at most $v^2/4$ edges.

7a) The degrees of the vertices (n-1) are even if and only if n is odd. Therefore there is an Euler circuit if and only if n is odd (and greater than 1).
7b) For all n >= 3, C_n has an Euler circuit (itself).
7c) Since the degrees of the vertices around the rim are all odd, no wheel has an Euler circuit.
7d) The degrees of the vertices are all n. Therefore there is an Euler circuit if and only if n is even (and greater than 0).

8a) Since each vertex connects to every other vertex, K_n has a Hamilton circuit for all n >= 3, but not for n <= 2
8b) The cycle itself is a Hamilton circuit.
8c) A Hamilton circuit for C_n can be extended to one for W_n by replacing one edge along the rim of the wheel with two edges, one going to the center and one leading away from the center. Therefore, W_n has a Hamilton circuit for n >= 3.

9) Prove that the graph representing the legal moves of a knight on an m x n chessboard, where m and n are positive integers, is bipartite. We will prove this by finding a method of partitioning the set of squares of the chessboard into two sets. Each square of the board can be thought of as a pair of integers (x,y). Let A be the set of squares for which x+y is odd, and let B be the set of squares for which x+y os even. This partitions the vertex set of the graph representing the legal moves of a knight into two parts. Every move of the knight changes x+y by an odd number ((x ± 2, y ± 1), (x ± 1, y ± 2) gives us x ± 2 + y ± 1 or x ± 1 + y ± 2; either 1+2 = 3, 1-2 = -1, -1+2 = 1, -1-2 = -2). Thus, every edge in this graph joins a vertex in set A to a vertex in set B; if x+y is odd, then odd + odd always gives an even, if x+y is even then even + odd always gives an odd. Thus, we have a bipartite graph, when we divide the set of squares of a chessboard as defined above.

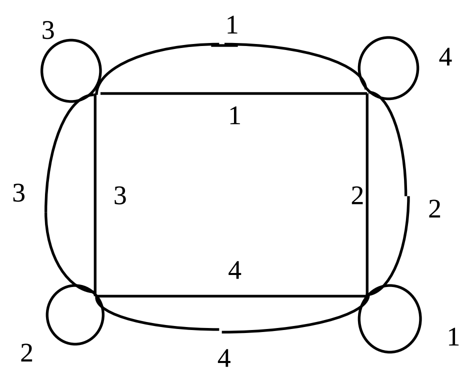
10) $B * B^T$ (i, j) = degree of i = in-degree + out-degree if i = j
= -(number of edges connecting i and j) if i != j

11) 0.68 seconds. The actual printing time is 0.58 seconds and the deadheading time is 0.10 seconds. There are six odd vertices. If we label them a through f clockwise from the top of the figure, we get a matching of minimum weight of: {a-d, b-c, e-f}. Note that the Chinese Postman Problem is defined in terms of circuits so we must start and end at the same vertex.

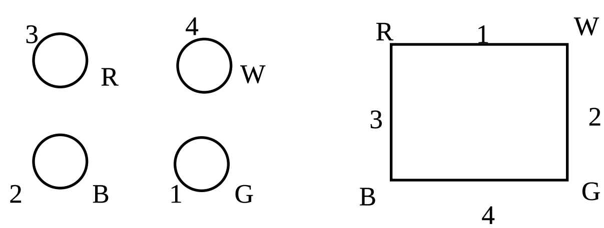
12) First answer is yes: consider a graph with 7 vertices numbered from 0 to 6, and edges corresponding to dominoes (edge from 2 to 4 for domino (2,4)); each vertex has degree 8, counting the loop. This means there exists an Euler circuit.

Second answer is no: now each vertex has degree 7.

13) The underlying graph is:



The acceptable subgraphs are:



So a solution is: