Hand In: (2.2) 7, 10, (2.3) 16, 28

Hint for 2.2.7: By symmetry, each edge of K_n appears in the same number of spanning trees of K_n.
Hint for 2.2.10: Only compute $\tau(K_{2,m})$

2.2.7

(!) Use Cayley's Formula to prove that the graph obtained from K_n by deleting an edge has (n - 2)n^{n-3} spanning trees.

Given n vertices, there are $2^{\binom{n}{2}}$ possible simple graphs. Using this, we may determine that there are n^{n-2} possible spanning trees (Cayley's Formula).
Since K_n contains the maximum possible amount of edges, K_n also contains the maximum possible amount of spanning trees for n vertices: n^{n-2}
Since there are n - 1 edges per spanning tree and $\frac{n(n-1)}{2}$ edges per complete graph, it makes sense to say that

$$\frac{(n-1)\times n^{n-2}}{\frac{n(n-1)}{2}}$$

represents the total number of spanning trees containing any edge in the graph.
Since n^{n-2} is the number of possible spanning trees in K_n, then

$$n^{n-2} - \frac{(n-1)\times n^{n-2}}{\frac{n(n-1)}{2}}$$

must be the total number of possible spanning trees in K_n - the total number of spanning trees that use a particular edge.
Starting with:

$$n^{n-2} - \frac{(n-1)\times n^{n-2}}{\frac{n(n-1)}{2}}$$

$$n^{n-2} - \frac{n^{n-2}}{\frac{n}{2}}$$

$$n^{n-2} - 2n^{n-3}$$

$$n \times n^{n-3} - 2n^{n-3}$$

$$(n-2)n^{n-3}$$

Hence, result.

2.2.10

Compute $\tau(\mathrm{K}_{2,m})$.

$\tau(\mathrm{K}_{2,m})$ = the number of spanning trees in G.

I find it easier to think of the tree as as a striped parallelogram (with one vertex from the 2 vertex partite on the left and the other on the right with the m vertices in the middle).

A spanning tree can be made made from choosing either the left or right vertex, including all the edges to the m partite set, and then picking one of the edges from the m partite set to the other vertex from the 2 partite set. So at this point we have m possible spanning trees. If we fix one minimal path from the left vertex to the right (which includes one vertex from the m set), then we have m - 1 leftover vertices in the m set. We can actually make a spanning tree from choosing any one of the two edges that go to the left vertex and right vertex to make our claw. So, given that we chose a particular minimal path from m objects, then, from the leftover m - 1 minimal paths, we can choose one edge or the other to make our claw, we have:
m2^{m-1} possible spanning trees.

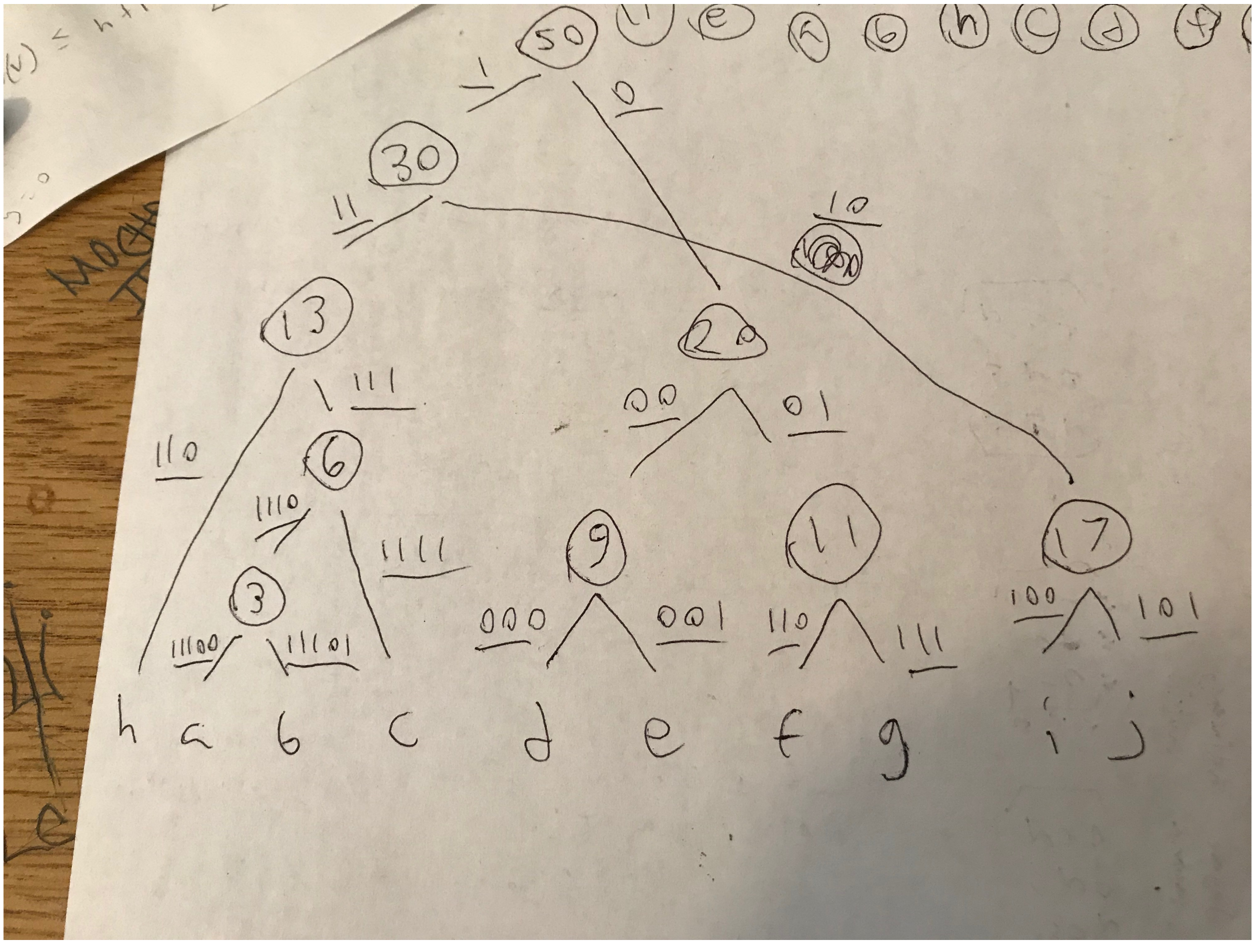
2.3.16

Four people must cross a canyon at night on a fragile bridge. At most two people can be on the bridge at once. Crossing requires carrying a flashlight, and there is only one flashlight (which can cross only by being carried). Alone, the four people cross in 10, 5, 2, 1 minutes, respectively. When two cross together, they move at the speed of the slower person. In 18 minutes, a flash flood coming down the canyon will wash away the bridge. Can four people get across in time? Prove your answer without using graph theory and describe how the answer can be found using graph theory.

Yes, it is possible. Just send the person who takes 2 minutes with the person who takes 1 minute (let's call these people by the number of minutes they take to cross the bridge), and then have 1 return with the flashlight. At this point, 3 minutes have elapsed.
Then, send 5 and 10 with the flashlight and have 2 return with it. At this point, 3 + 10 + 2 = 15 minutes have elapsed.
Finally, have 1 and 2 cross the bridge. At this point, 17 minutes have elapsed. They have crossed the bridge in time.
In terms of graph theory, if we think of the amount of time it takes to cross the bridge as a weight for the edge between two vertices, then we just have to come up with all possible combinations of states this scenario could have, and represent them as vertices.
Then, if it's possible to get from one state to another, we draw an edge between each state's vertex and put a weight corresponding to the amount of time between each state change.
Finally, we're trying to get from some initial state to the ending state in the shortest time possible, so it makes sense to use a shortest path algorithm (like Dijkstra's) to find the shortest path.

2.3.28

Compute a code with minimum expected length for a set of ten messages whose relative frequencies are 1, 2, 3, 4, 5, 5, 6, 7, 8, 9. What is the expected length of a message in this optimal code?



item	frequency	code	length
a	1	11100	5
b	2	11101	5
c	3	1111	4
d	4	000	3
e	5	001	3
f	5	110	3
g	6	111	3
h	7	110	3
i	8	100	3
j	9	101	3

$$1+2+3+4+5+5+6+7+8+9 = 50 \text{ occurrences.}$$

$$\text{Expected Length: } \frac{1}{50}5+\frac{2}{50}5+\frac{3}{50}4+\frac{4}{50}3+\frac{5}{50}3+\frac{5}{50}3+\frac{6}{50}3+\frac{7}{50}3+\frac{8}{50}3+\frac{9}{50}3 = 3.18$$