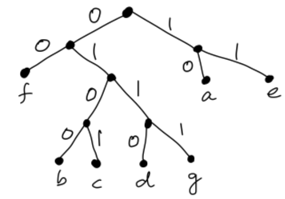
M115 Homework12, Spring 2022.



1. Given the character frequencies

$$(a, b, c, d, e, f, g) = (0.26, 0.02, 0.09, 0.08, 0.30, 0.14, 0.11)$$

I tried encoding the characters in a binary tree shown on the right.

a) With this encoding, what is the bit-string translation of the word 'bead' ?

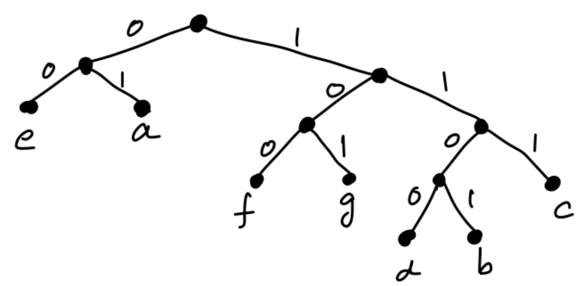
b) With this encoding, what is the average number of bits used per character?

c) If possible, find an encoding that has a lower average number than your answer to part (b).

a) 'bead' = 0100+11+10+0110 = 010011100110 (by '+' I do concatenation of the individual bit strings.)

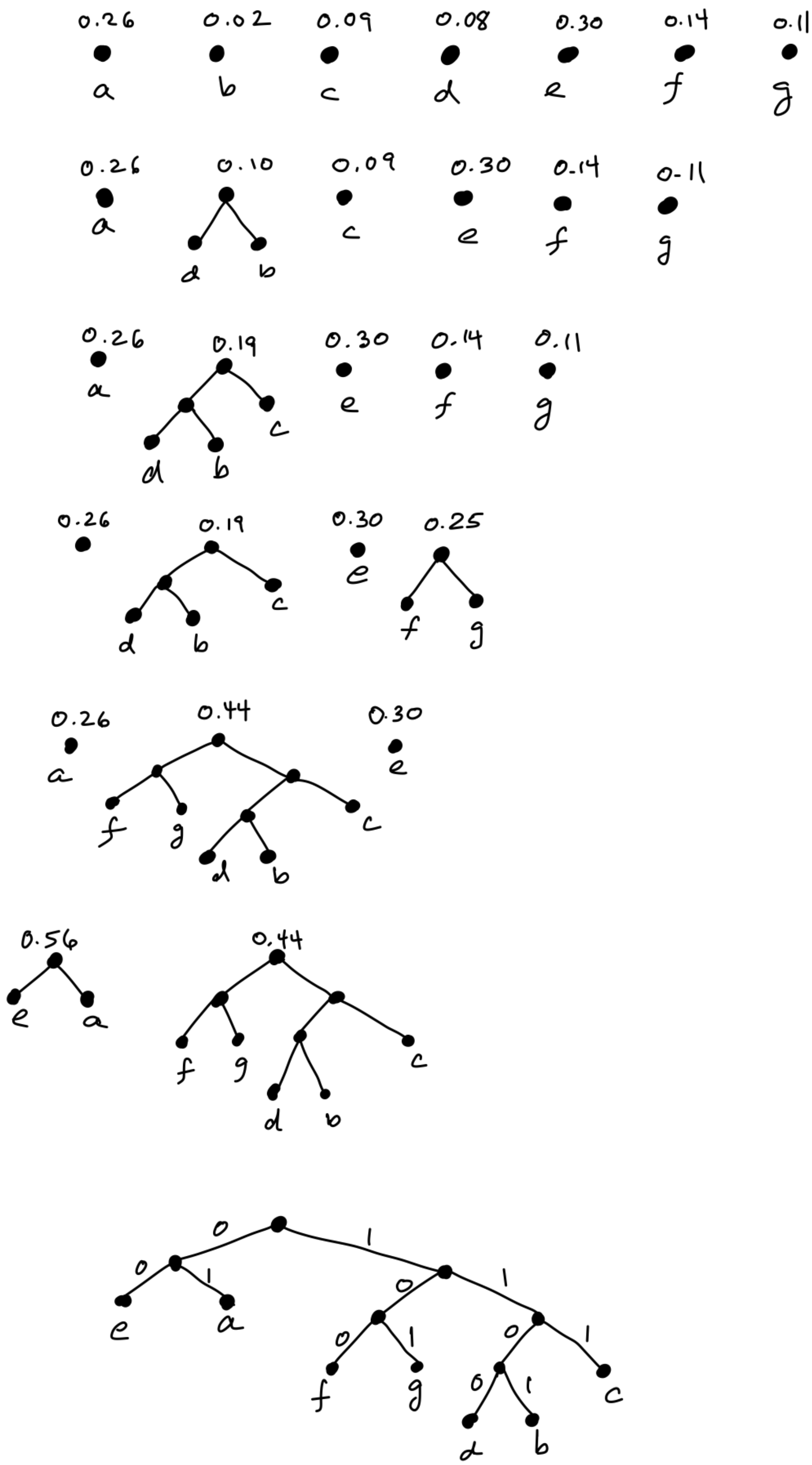
b) 2(0.14+0.26+0.30) + 4(0.02+0.09+0.08+0.11) = 2(0.7)+4(0.3)=2.6

c) Use Huffman coding.
I got an encoding given by the binary tree on the right. (my work is on the next page.)



This has an average number of bits per character
2(0.26+0.30)+3(0.09+0.14+0.11)+4(0.02+0.08)=
2(0.56)+3(0.34)+4(0.10)=2.54.

Here are the details to the Huffman coding in question 1c.



Also, from the online textbook, do problems:

12.23, 12.56ab, 12.57, 12.63.

optional extra credit: 12.54; compute the chromatic polynomial of the graph in 12.23

In question 12.56ab you only need to work through the two algorithms (Kruskal and Prim), you don't need to work through the problems' request regarding black-white colorings and grey edges.

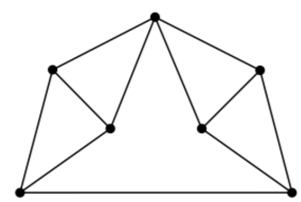
If you don't want to draw lots of pictures, you may show your work in each part by clearly indicating the order in which edges are included (if at all) in a minimum spanning tree.

Problem 12.23

Problems for Section 12.6

Class Problems

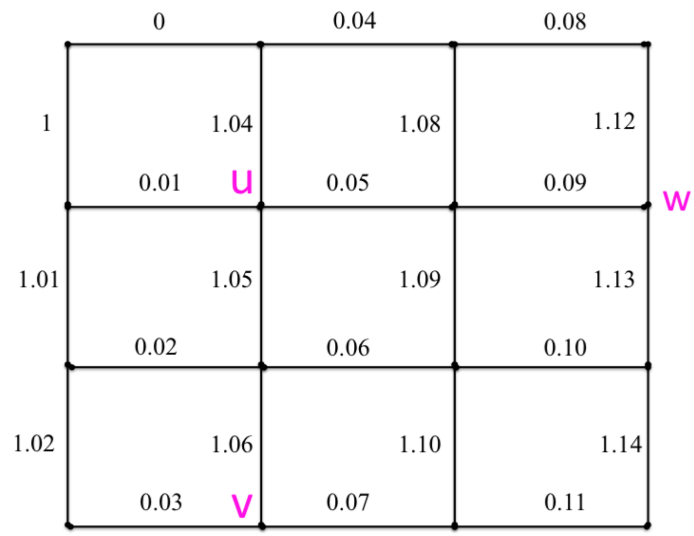
Problem 12.23.
Let G be the graph below.[16] Carefully explain why $\chi(G) = 4$.



To show that the chromatic number is 4, we can argue that the graph cannot be colored with only 3 colors, but it can with 4 colors. Remember 'coloring' the graph means coloring the vertices so that no two adjacent vertices are the same color.
Suppose we only had three colors, say {red, green, blue}. The two vertices at the bottom of the graph are adjacent, so they need different colors, wlog the bottom right vertex is blue, and the bottom left vertex is green.
The four vertices in the middle of the graph, the ones with degree 3 which haven't been colored yet, they will need to receive all three colors. The two middle vertices on the right, being adjacent to the bottom right vertex, will be two colors that aren't blue, so red and green; and the two middle vertices on the left will be two colors that aren't green, so red and blue.
The top vertex is in trouble. Why are you going to color it? It's adjacent to four other vertices which have all been colored with the three different colors. Hence there are no choices for the color of that top vertex when you only have three colors.
If there were four colors, say {red, green, blue, yellow}, then there is a coloring: color the bottom and middle vertices as before, then color the top vertex yellow.
So 4 is the smallest number of colors which admits a graph coloring, so 4 is the chromatic number.

Problem 12.56

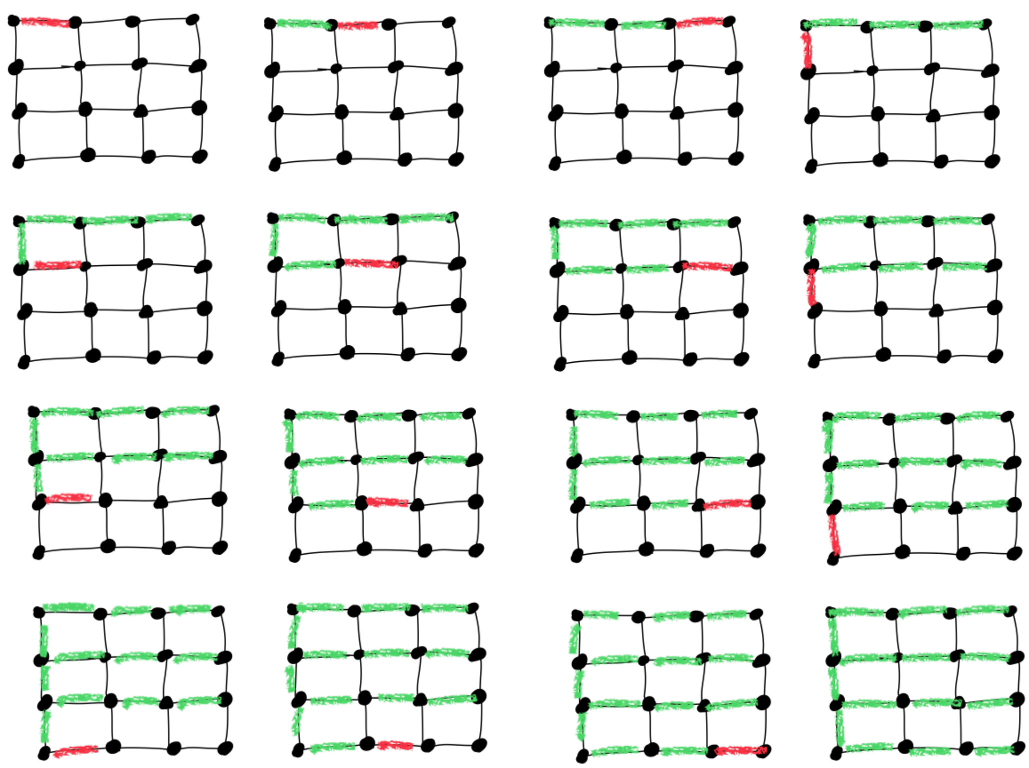
I'll label the edges according to their weights. None of the weights are repeated, so the weights are a good indication of the edges in this graph



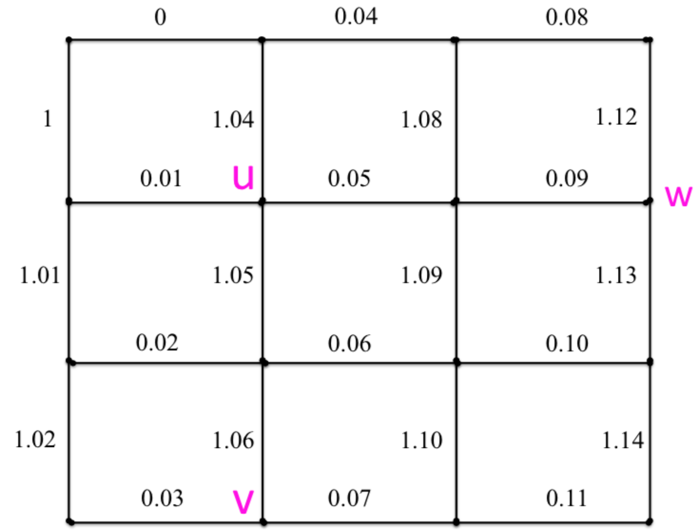
a) Using Prim's Algorithm

edges: 0, 0.04, 0.08, 1, 0.01, 0.05, 0.09, 1.01, 0.02, 0.06, 0.10, 1.02, 0.03, 0.07, 0.11

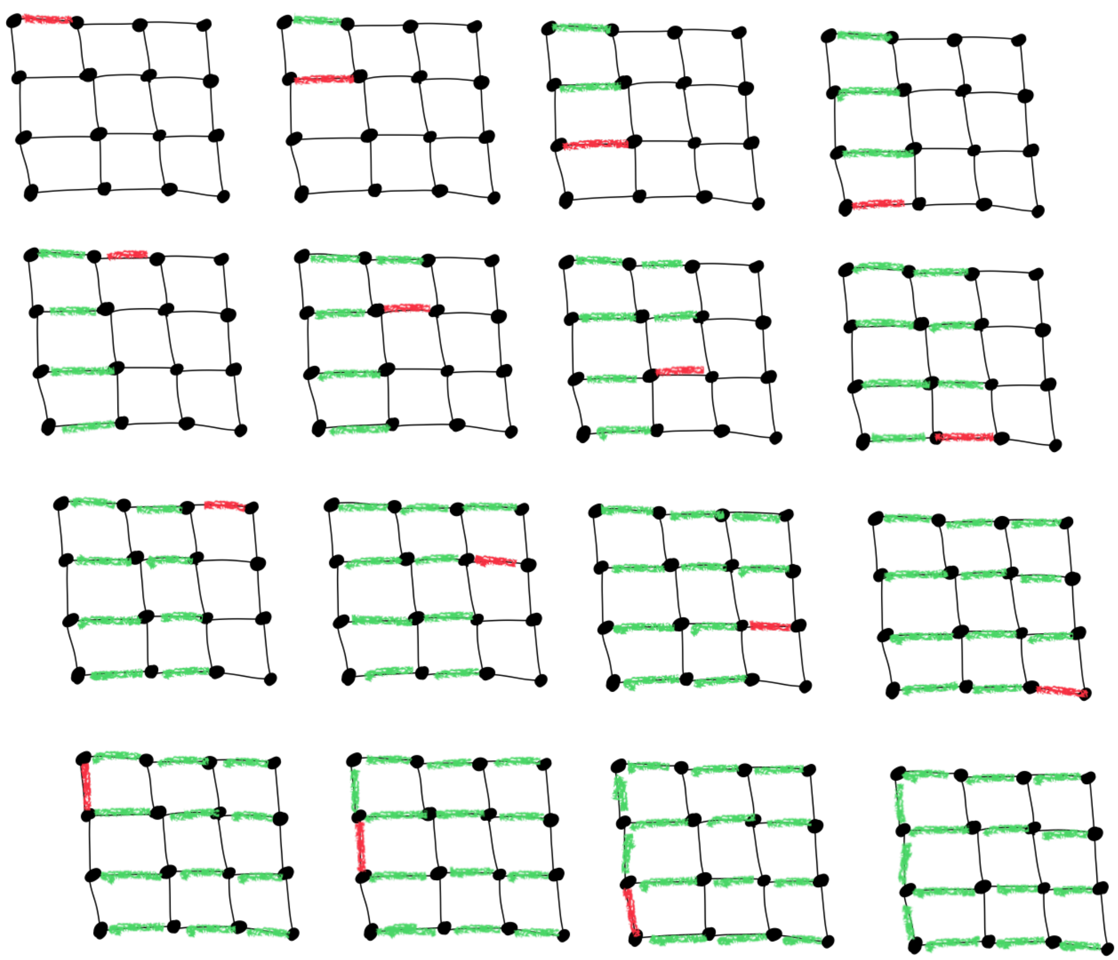
Prim's algorithm starts with a small tree which grows larger and larger until it spans the given graph:



b) Using Kruskal's Algorithm
edges: 0, 0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.10, 0.11, 1, 1.01, 1.02



Kruskal's Algorithm features a forest of trees which grow ever more interconnected until you have a single tree that spans the whole graph:



Problem 12.57.

In this problem you will prove:

Theorem. *A graph G is 2-colorable iff it contains no odd length closed walk.*

As usual with "iff" assertions, the proof splits into two proofs: part (a) asks you to prove that the left side of the "iff" implies the right side. The other problem parts prove that the right side implies the left.

(a) Assume the left side and prove the right side. Three to five sentences should suffice.

(b) Now assume the right side. As a first step toward proving the left side, explain why we can focus on a single connected component H within G.

(c) As a second step, explain how to 2-color any tree.

(d) Choose any 2-coloring of a spanning tree T of H. Prove that H is 2-colorable by showing that any edge *not* in T must also connect different-colored vertices.

Problem 12.57

a) Assume we have an (undirected) graph G which is 2-colorable. We want to show if has no closed walks of odd length. We can try arguing by contradiction. Suppose it <u>does</u> have a closed walk of odd length, like this $vu_1u_2 \ldots u_nv$. This walk having odd length means n is even, so there are an even number of vertices aside from the start and end vertex v.

Since the graph is 2-colorable, we can color the vertices with two colors, say red (r) and blue (b) so that no two adjacent vertices have the same color. In particular the vertices on this walk have alternating colors, like this, assuming wlog that v is colored red: $vu_1u_2 \ldots u_nv =$ rbrb…?r. But if n is even, then the color 'r' and 'b' will occur in pairs among the vertices $u_1, u_2, \ldots, u_n$, so the color of u_n would be r, which is a contradiction since u_n is adjacent to v.

Thus if the graph is 2-colorable, it can't have any closed walks of odd length.

b) Now we're assuming that G has no odd-length walks, and we want to show that G is 2-colorable.

If G is not connected (so it has many connected components), then we can try to color the first component, then the second, and so on. Whatever method we use for coloring the first component will work for other components just as well since the assumption about the walks applies to each component individually.

12.57 c)

One feature of a tree is that there are unique paths between any two vertices. This can be helpful in coloring a tree with two colors.

Say we have colors {red, blue} = {r,b}. Start with your favorite vertex in the tree, u. Color it red. Suppose v is another vertex in the tree. Since the graph is a tree there is a unique path from u to v. If that path has an odd length (i.e. and odd number of edges), then color v blue. If the path has an even length, then color v red.

Does this result in a proper coloring of the graph? If not, then somewhere there will be adjacent vertices that have the same color. Call these vertices v_1, v_2. There is a unique path from vertex u to v_1, call it $u a_1 a_2 a_3 \ldots a_n v_1$. There is also a unique path from u to v_2, call it $u b_1 b_2 b_3 \ldots b_m v_2$. Since v_1 and v_2 are adjacent the unique path to one of these vertices must go through the other vertex otherwise you could make a cycle, contrary to the assumption that we have a tree. Assuming wlog that the path to v_2 is the longer of the two paths, we would need to use the edge $\{v_1, v_2\}$ in the path from u to v_2, so the path from u to v_2 would actually traverse first the path from u to v_1 followed by the edge from v_1 to v_2. In other words, you could say $u b_1 b_2 b_3 \ldots b_m v_2 = u a_1 a_2 a_3 \ldots a_n v_1 v_2$. But then the lengths of the path from u to v_2 has length that is one more than the length of the path from u to v_1, so these two paths can't possibly have the same parity: one of the paths will have even length, the other one will have odd length. The contradicts the assumption that v_1 and v_2 have the same color. Thus the coloring scheme does result in a graph coloring: adjacent vertices will receive different colors.

12.57 d)

We have a spanning tree T of a connected graph H (since H is one of the connected components of the original graph G). We can color the tree T with two colors as described in part (c). Since T is a spanning tree of H, every single vertex of H is in T. So by coloring the tree T we have colored every vertex in the graph H. Whether or not this amounts to a coloring of the graph H is another question, we need to make sure that two vertices which are adjacent in H don't have the same color.

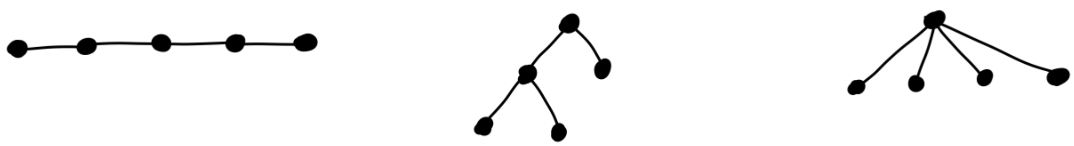
For the sake of contradiction, let's suppose this did happen: we colored the spanning tree T and while we don't have vertices adjacent in T with the same color, we do have vertices adjacent in H with the same color. We are still working under the assumption that H has no closed walks of odd length. Will there be a contradiction with this assumption somehow?

Say we have vertices v_1 and v_2 which are adjacent in H (but not in T), which have the same color from the coloring of T. According the the coloring of T we have that v_1 and v_2 have a unique path to a vertex u and that these paths are either both odd (case1) or both even (case2).
(Note: it is possible that u equals one of the vertices v_1, v_2, in which case one path has length 0, and the other path has even length.)
In both cases 1 and 2 we get a closed walk in the graph H by first traversing the path from u to v_1, then traversing the edge $\{v_1, v_2\}$ in the graph H, then traversing the path u to v_2 backwards, back to the vertex u. In both cases this closed walk has odd length, which contradicts the assumption that the graph G have no such walks.

Thus the coloring with two colors of the spanning tree T in the connected component H amounts to a valid coloring of H. We can repeat this process if necessary for the other connected components of G to get a valid coloring with two colors for the entire graph G. QED.

Problem 12.63.
How many 5-vertex non-isomomorphic trees are there? Explain.

Problem 12.63 I think there are three. Such a tree will be isomorphic to one of these:



If you want a tree to have five vertices, then there must be four edges. You can consider the possible degrees of the vertices. The sums of the degrees will be twice the number of edges by the handshaking theorem. So $d_1 + d_2 + d_3 + d_4 + d_5 = 8$, where d_i is the degree of the i^{th} vertex. Since a tree is connected, there can't be any isolated vertices, so each vertex must have degree at least 1, i.e. the quantities d_i are strictly positive.

Subtract 5 from both sides of the equation and we get $\tilde{d}_1 + \tilde{d}_2 + \tilde{d}_3 + \tilde{d}_4 + \tilde{d}_5 = 3$ where now the quantities $\tilde{d}_i = d_i - 1$ are merely non-negative instead of strictly positive; they might be 0. With non-negative summands, how many solutions does the equation $\tilde{d}_1 + \tilde{d}_2 + \tilde{d}_3 + \tilde{d}_4 + \tilde{d}_5 = 3$ have?

0+0+1+1+1 = 3
0+0+0+1+2 = 3
0+0+0+0+3 = 3

Those three equations encompass all the possibilities, up to the order of the summands. Adding one back to each of the summands, so adding 5 to both sides of the equations, we get vertex degrees for three possible graphs:

1+1+2+2+2=8 (this is the sum of the vertex degrees for the graph above on the left)
1+1+1+2+3=8 (this is the sum of the vertex degrees for the graph above in the middle)
1+1+1+1+4=8 (this is the sum of the vertex degrees for the graph above on the right)