CS 527
Tenghuan Li
HW2
liten@oregonstate.edu

Problem1:

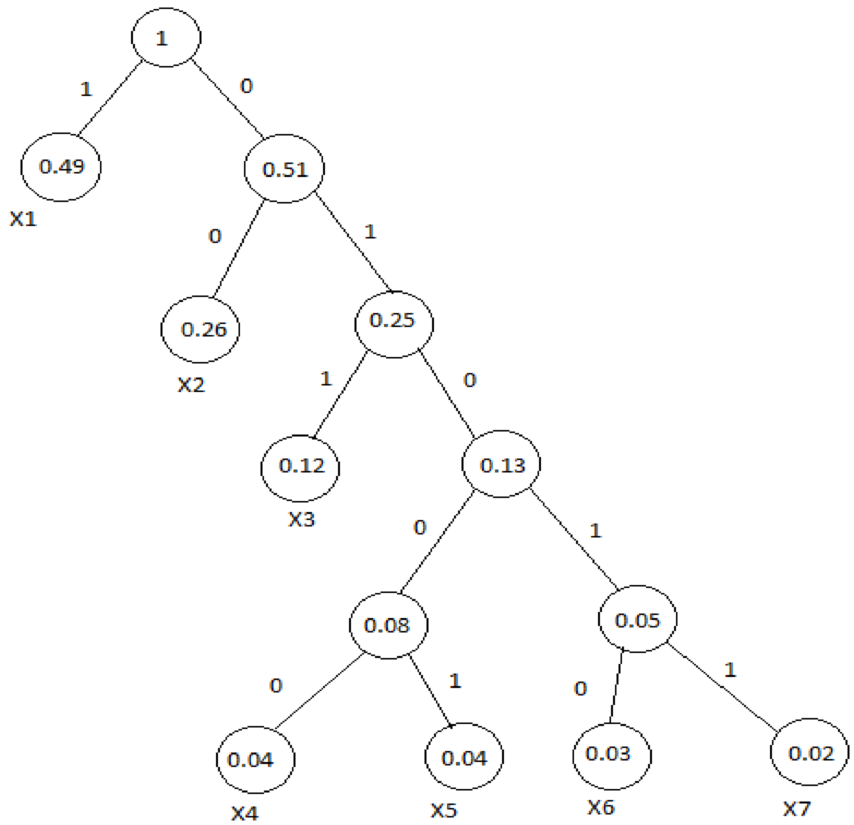
a. No, because 0 is a prefix for 01.
b. Yes, because C(001) = C(0)C(1).
c. Yes, because it is uniquely to decodable.

Problem2:

As the question we know: $X = [\frac{X1 \quad X2 \quad X3 \quad X4 \quad X5 \quad X6 \quad X7}{0.49\ 0.26\ 0.12\ 0.04\ 0.04\ 0.03\ 0.02}]$

a.

| X | Codeword | Length |
|---|---|---|
| X1 | 1 | 1 |
| X2 | 00 | 2 |
| X3 | 011 | 3 |
| X4 | 01000 | 5 |
| X5 | 01001 | 5 |
| X6 | 01010 | 5 |
| X7 | 01011 | 5 |



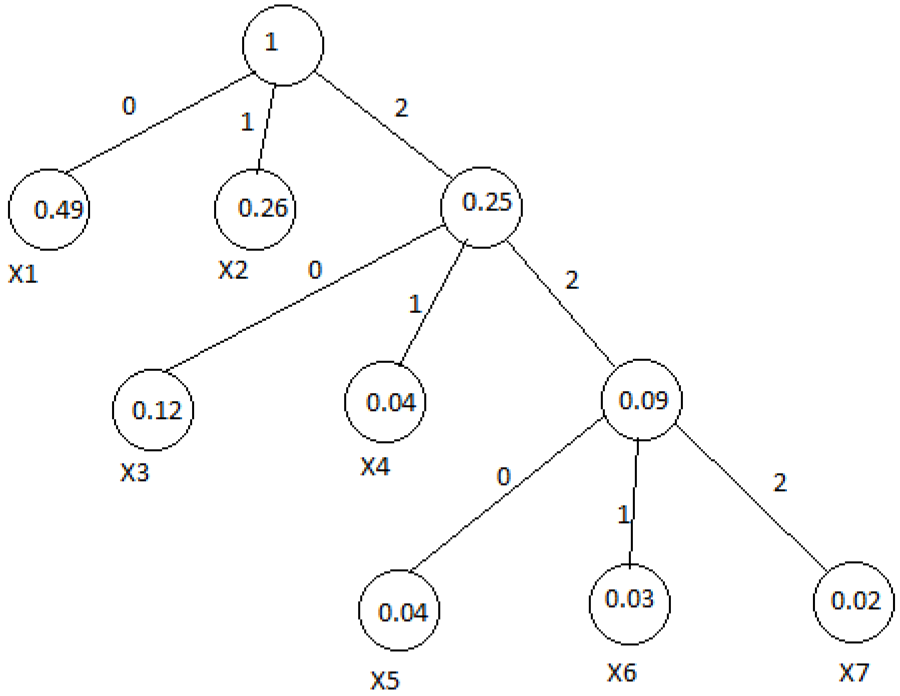
b. So as the table and picture in a part.

$$Lav = \sum_i P_i l_i = (1 * 0.49 + 2 * 0.26 + 3 * 0.12 + 5 * 0.04 + 5 * 0.04 +$$
$$5 * 0.03 + 5 * 0.02)$$

= 2.02 bits

c.

| X | Codeword | Length |
|---|---|---|
| X1 | 0 | 1 |
| X2 | 1 | 1 |
| X3 | 20 | 2 |
| X4 | 21 | 2 |
| X5 | 220 | 3 |
| X6 | 221 | 3 |
| X7 | 222 | 3 |




$$Lav = \sum_i P_i l_i = (1 * 0.49 + 1 * 0.26 + 2 * 0.12 + 2 * 0.04 + 3 * 0.04 +$$
$$3 * 0.03 + 3 * 0.02)$$

= 1.34 bits

Problem 3:

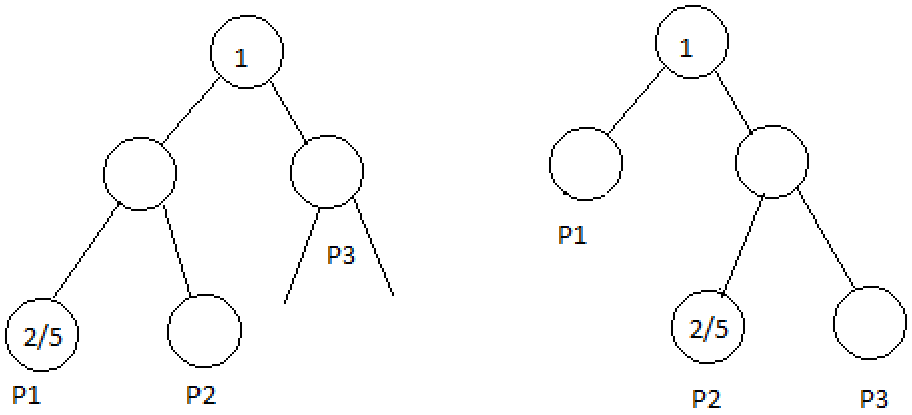
a. {0, 10, 11}.

Yes, this code could be a Huffman code because it achieves the 3 properties of the Huffman codes.

b. {00, 01, 10, 110}.
   No, this violates the 2nd property of the Huffman code because the two longest codewords have the same length. So, the answer is no.
c. {01, 10}
   No, because we do the optimize to get the average length {0,1}

Problem 4:

a. As the problem, $P_1 > 2/5$, $l_1 = 1, P_1 > P_2 \geq P_3 \geq \cdots \geq P_m$.
   We need to prove this by Contradiction.
   Assume $l_1 \geq 2$, Now the Huffman tree looks like Pa.




(Pa) (Pb)

Since $P_3 > P_1, P_3 > P_2$ we can improve the Huffman code by rearranging the tree so that the word lengths are ordered by increasing length from top to bottom.

Then, we swap probability the assignments to improve the expected depth of the tree as the Pb shows.

Then, $l_1 = 1$ that contradicts our assumption. So $l_1$ *must be* 1.

b. $P_1 < {}^{1}/_{3} \rightarrow l_1 \geq 2$

   Assume $l_1 = 1$. In the other node, it is a symbol with probability > 1/3, call $P_c$ and $P_c > P_1$, $P_1$ is other symbol in the Huffman tree. In conclusion, this contradicts our assumption ($l_1 = 1$). So $l_1 \geq 2$.

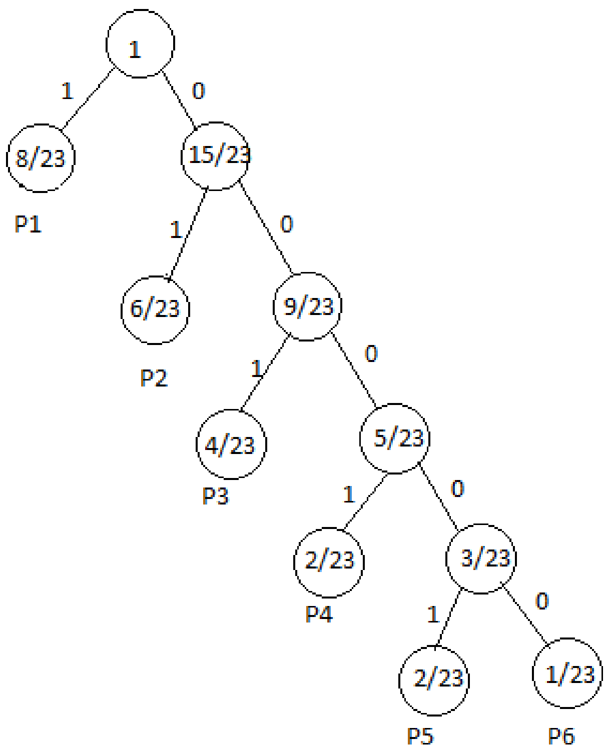
Problem 5:

a. Because expected number of tastings $\equiv Lav$.

   Then, $Lav = \sum_i P_i l_i = \left(1 * \frac{8}{23} + 2 * \frac{6}{23} + 3 * \frac{4}{23} + 4 * \frac{2}{23} + 5 * \frac{2+1}{23}\right) = \frac{55}{23}$
   = 2.391 bits

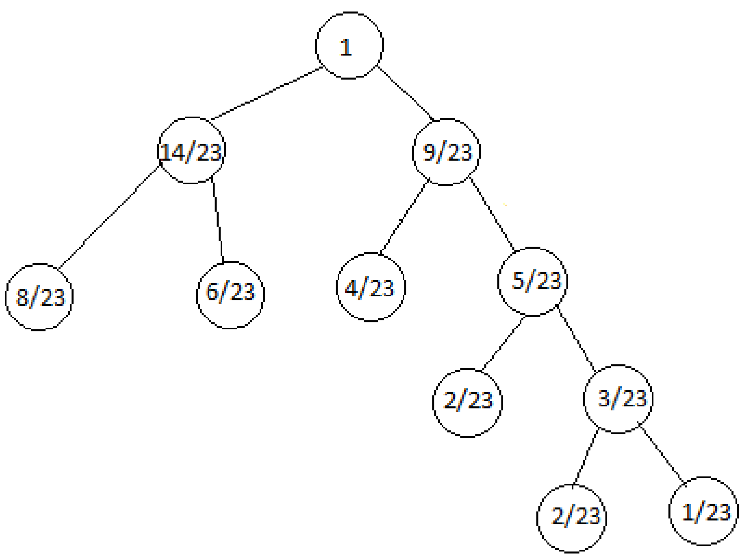
b. The first bottle to be tasted should be the one with highest Prob $\frac{8}{23}$.

   Because as the Huffman tree shows, at Picture C.




(Picture C)

c. As the tree shows:




Then, $Lav = \sum_i P_i l_i = \left(2 * \frac{8}{23} + 2 * \frac{6}{23} + 2 * \frac{4}{23} + 3 * \frac{2}{23} + 4 * \frac{2+1}{23}\right) = \frac{54}{23}$

$= 2.3478$ bits or tastings

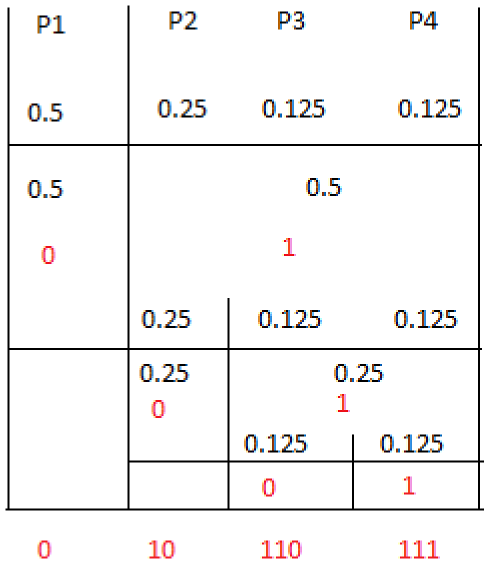
Time of tasting $\equiv$ length of the code.

Problem 6:

a. Because the code for the probability distribution is (0.5, 0.25, 0.125, 0.125).

So, and $l_i = \left\lceil \log \frac{1}{p_i} \right\rceil$ we can get the table like:

| Symbol | Pi | Fi(decimal) | Fi(binary) | $l_i$ | codeword |
|---|---|---|---|---|---|
| X1 | 0.5 | 0.0 | 0.0 | 1 | 0 |
| X2 | 0.25 | 0.5 | 0.10 | 2 | 10 |
| X3 | 0.125 | 0.75 | 0.110 | 3 | 110 |
| X4 | 0.125 | 0.875 | 0.111 | 3 | 111 |




b. $H(x) \leq L < H(x) + 1$

We need to probe the both bounds:

First of all, $H(x) \leq L$:

$l_i = \left\lceil \log \frac{1}{p_i} \right\rceil \geq \log_2 {}^1/_{p_i} \quad \rightarrow * \; p_i(\text{when } \; p_i \geq 0, \; \text{it doesn't change})$

$P_i l_i \geq P_i \log_2 {}^1/_{p_i},$

So, $\sum_i P_i l_i \geq \sum_i P_i \log_2 {}^1/_{p_i} \Rightarrow Lav \geq H(x)$. ................................1

So, we can get $Lav \geq H(x)$.

Then, $L < H(x) + 1$:

$l_i = \left\lceil \log \frac{1}{p_i} \right\rceil < \log_2 {}^1/_{p_i} + 1$

$P_i l_i < P_i(\log_2 {}^1/_{p_i} + 1)$

$$\sum_i P_i l_i < \sum_i P_i(\log_2 {}^1/_{p_i} + 1) \Rightarrow Lav < \sum_i P_i \log {}^1/_{p_i} + \sum_i P_i \Rightarrow Lav < H(x) + 1$$

So, we can get $Lav < H(x) + 1$.........................................................2

make this two together, it is $H(x) \leq L < H(x) + 1$.

Then we need to prove the prefix free property.

Because $l_i = \left\lceil \log \frac{1}{p_i} \right\rceil$

$$l_i \geq \log_2 \frac{1}{p_i}$$

$$\frac{1}{l_i} \leq \frac{1}{\log_2 \frac{1}{p_i}}$$

$$\frac{1}{2^{l_i}} \leq \frac{1}{2^{\log_2 \frac{1}{p_i}}}$$

So, we can get $2^{-l_i} \leq P_i$ .

Now, $F_i$ different form $F_i$ by at $2^{-l_i} \leq P_i$ , and this is clear from the right-side figure. This is in terms of the value.

Then, in this term of codewords, $F_i$ codeword is different with $F_i$ codeword in once place at the first $l_i$ places. So, when all the intervals of $F_i$ are disjoint, this ensures the prefix tree property of the set of the codewords.

$F_3 = 0.75, P_3 = 0.125, F_4 = 0.875,$

$$P_3 = 0.125 = \begin{cases} F_4 = 0.875 = 0.001 + 0.110 = 0.111 \\ \qquad\qquad F_3 = 0.75 = 0.1101 \end{cases}$$

= 0.001

In conclusion, we can know the $F_4$ is different with $F_3$ by last place.