ATHARVA DESHPANDE
deshpana@oregonstate.edu
CS 527
HW 2

1.

a) Binary code with lengths 2, 2, 2, 3, 3.

By Kraft's Inequality, tree code must satisfy $\sum_{i=1}^{n} D^{-l_i} \leq 1$.

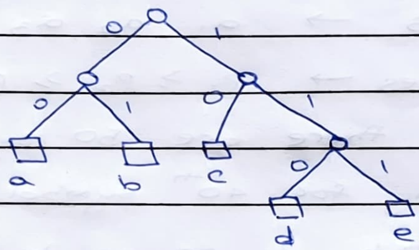
Here, $D = 2$ and $l_i = 2, 2, 2, 3, 3$.

$\therefore \sum_{i=1}^{n} D^{-l_i} = \frac{1}{2^2} + \frac{1}{2^2} + \frac{1}{2^2} + \frac{1}{2^3} + \frac{1}{2^3}$

$= \frac{1}{4} + \frac{1}{4} + \frac{1}{4} + \frac{1}{8} + \frac{1}{8} = 1 \leq 1$.

$\therefore$ Tree code is possible.

Tree Code for the above problem:



a → 00, b → 01, c → 10, d → 110, e → 111.

b) Ternary code with lengths 1, 2, 2, 3, 3, 3, 3.

By Kraft's Inequality, tree code must satisfy $\sum_{i=1}^{n} D^{-l_i} \leq 1$.

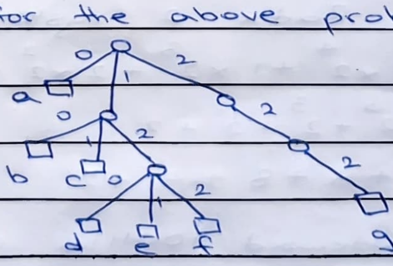
Here, $D = 3$ and $l_i = 1, 2, 2, 3, 3, 3, 3$.

$\therefore \sum_{i=1}^{n} D^{-l_i} = \frac{1}{3^1} + \frac{1}{3^2} + \frac{1}{3^2} + \frac{1}{3^3} + \frac{1}{3^3} + \frac{1}{3^3} + \frac{1}{3^3}$

$= \frac{1}{3} + \frac{1}{9} + \frac{1}{9} + \frac{1}{27} + \frac{1}{27} + \frac{1}{27} + \frac{1}{27} = \frac{19}{27} = 0.704 \leq 1$.

$\therefore$ Tree code is possible.

Tree Code for the above problem:



a → 0, b → 10, c → 11, d → 120, e → 121, f → 122, g → 222

c) Ternary Code with lengths 1, 1, 2, 2, 2, 3.

By Kraft's Inequality, tree code must satisfy

$\sum_{i=1}^{n} D^{-l_i} \leq 1$.

Here, $D = 3$ and $l_i = 1, 1, 2, 2, 2, 3$.

$\therefore \sum_{i=1}^{n} D^{-l_i} = \frac{1}{3^1} + \frac{1}{3^1} + \frac{1}{3^2} + \frac{1}{3^2} + \frac{1}{3^2} + \frac{1}{3^3}$

$= \frac{1}{3} + \frac{1}{3} + \frac{1}{9} + \frac{1}{9} + \frac{1}{9} + \frac{1}{27} = \frac{28}{27} = 1.04 > 1$.

$\therefore$ Tree Code is not possible.

2. n symbols $\rightarrow$ $a_1, a_2, a_3, \ldots, a_n$.

Probabilities $\rightarrow$ $P_1 \geq P_2 \geq P_3 \geq \ldots \geq P_n$

a) Assuming there are 50 symbols, codeword corresponding for 40^{th} symbol a_{40}:

Dividing 40 by 2 so as to keep remainder $\{2, 1\}$

$$\begin{array}{r|ll} 2 & 40 & \\ 2 & 19 & \rightarrow 2 \\ 2 & 9 & \rightarrow 1 \\ 2 & 4 & \rightarrow 1 \\ 2 & 1 & \rightarrow 2 \\ & 0 & \rightarrow 1 \end{array}$$

i.e. 12112.

Now, subtracting 1 from all bits;

$\therefore a_{40} = \boxed{01001}$ Answer.

b) codeword = 100111s.

Adding 1 to all bits; $\Rightarrow$ 211222

Now, using Radix-2 system;

$2 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 2 \times 2^2 + 2 \times 2^1 + 2 \times 2^0$

$= 2^6 + 2^4 + 2^3 + 2^3 + 2^2 + 2$

$= 64 + 16 + 8 + 8 + 4 + 2 = 102$.

Hence, the codeword 100111s corresponds to the 102^{nd} symbol a_{102}.

c) 16 symbols each with probabilities $1/16$.

Symbols	a_1	a_2	a_3	....	a_{16}
Probability	$1/16$	$1/16$	$1/16$	....	$1/16$

Considering binary codewords as 0s, 1s, 00s, ...

Average Length $L_{av} = \sum_{i=1}^{16} P_i l_i$

$$= \frac{1}{16}\left[1+1+2+2+2+2+3+3+3+3+3+3+3+3+4+4\right] + 1$$

(for space)

$$= \frac{1}{16}(42) + 1 = \frac{29}{8}$$

$$= 3\left(\frac{5}{8}\right) = 3.625 \text{ bits}.$$

We know, $H(x) = \sum_{i=1}^{n} P_i \log 1/P_i$

$$= \frac{1}{16}\log 16 + \frac{1}{16}\log 16 + \dots + \frac{1}{16}\log 16$$

[16 times]

$$= \log_2 16 = \log_2 2^4$$

$$= 4 \text{ bits}.$$

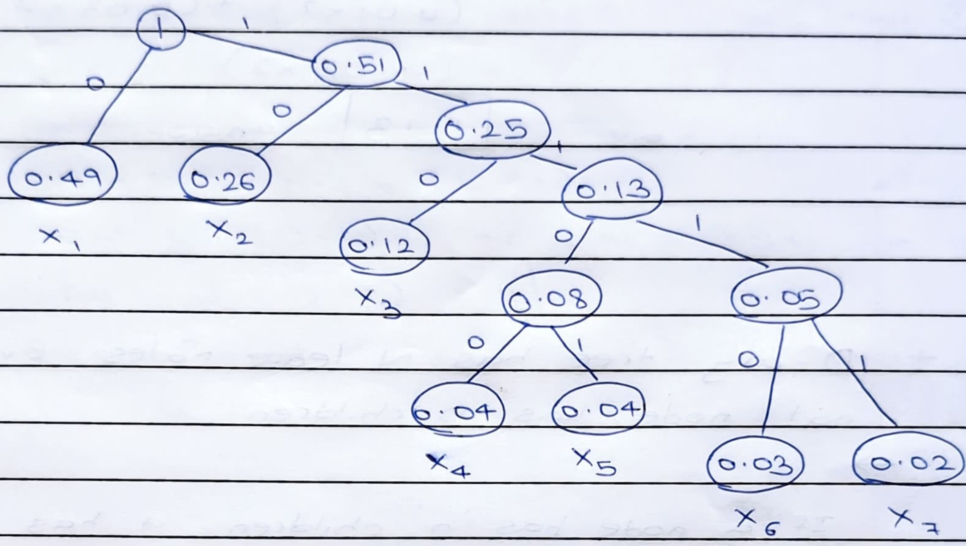
As we can see, there is some difference when compared with Huffman code.

In general, symbols in the above problem are assigned codewords sequentially without regard to their probabilities.

In Huffman Coding, codewords of symbols ~~with~~ are determined based on their probabilities; which in turn provides better compression.

3. $X = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 \\ 0.49 & 0.26 & 0.12 & 0.04 & 0.04 & 0.03 & 0.02 \end{bmatrix}$

a) Binary Huffman Code for X.



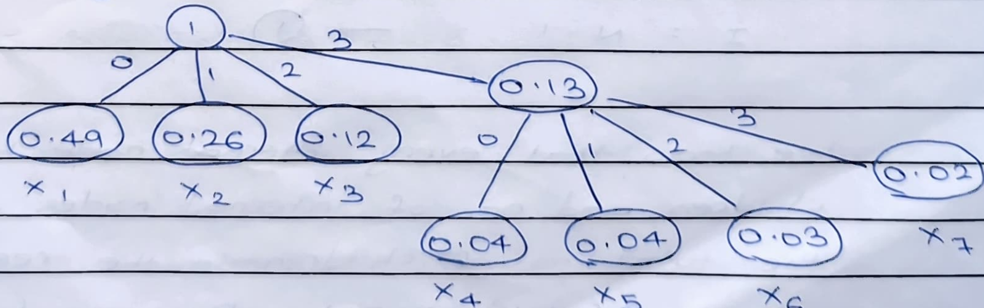
$x_1 \to 0$, $x_2 \to 10$, $x_3 \to 110$, $x_4 \to 11100$,
$x_5 \to 11101$, $x_6 \to 11110$, $x_7 \to 11111$.

b) Expected Code Length :

$$L_{av} = \sum_{i=1}^{n} P_i l_i = (0.49)\times 1 + (0.26)\times 2 + (0.12)\times 3 + (0.04)\times 5 + (0.04)\times 5 + (0.03)\times 5 + (0.02)\times 5$$

$$\therefore L_{av} = \boxed{2.02} \text{ Answer}$$

c) 4-ary Huffman Code for X.



$x_1 \to 0$, $x_2 \to 1$, $x_3 \to 2$, $x_4 \to 30$,
$x_5 \to 31$, $x_6 \to 32$, $x_7 \to 33$

Average length $= L_{av} = \sum_{i=1}^{n} P_i l_i$

$= (0.49 \times 1) + (0.26 \times 1) +$
$(0.12 \times 1) + (0.04 \times 2) +$
$(0.04 \times 2) + (0.03 \times 2) +$
(0.02×2)

$\therefore L_{av} = \boxed{1.13}$ Answer

4. D-ary tree has N leave nodes. Every internal node has D children.

→ If a node has 'n' children, it has 'n+1' subtrees.
Root of the tree has 'D' subtrees, one for each child. Each of these subtrees is a D-ary tree → Number of leaf nodes in each subtree is related.

$\therefore$ If total number of leaf nodes in a D-ary tree is N, and let's assume total number of internal nodes is I, then :

$N = I + 1$

$\therefore I = N - 1$ — ①

We know that every internal node has D children and no. of internal nodes is I, so the total no. of children in the tree is :

$DI = D(N-1) = DN - D$ — (From ①)

Now, let's consider D-ary tree as a full tree (since the question mentions every internal node).

$\therefore$ Total no. of children

= Total no. of internal nodes + no. of leaf nodes

$\therefore \quad DN - D = N + (D-1) * I$

$\therefore \quad DN - D = N + (D-1)(N-1) \quad$ — (From ①)

$\therefore \quad DN - D = N + (D-1)N - (D-1)$

$\therefore \quad \cancel{DN - D = N + DN - N - D + 1}$

$\therefore \quad DN - N = DN - (D-1)$

$\therefore \quad -N = -(D-1) \quad$ or $\quad N = D - 1$

$\therefore \quad \boxed{N \bmod (D-1) = 1}$

Hence Proved.

5. Assuming tree to have atleast 2 levels, weights of the level 2 nodes are a, b, c, d.
Without loss of generality, assuming A and B are joined first, then C and D.
So, $a, b \leq c, d \leq a+b$.

a) If $p_0 > 3/5$ or in this case a or $b > 3/5$, then c and d must be greater than $3/5$ leading to impossible scenario.
So, no codeword of length 2 or greater can be formed.
Contrapositively, if there is codeword for p_0 of length 2 or greater, then a and b both must be $\leq 3/5$.
Considering case where if $a > 3/5$, then there would be no codeword of length 2 or greater
$\therefore$ If $a > 3/5$, then its codeword must be of length 1.
Hence, If $p_0 > 3/5$, then corresponding symbol will have codeword of length 1.

b) Considering similar ~~scenari~~ scenario for a tree with at least 2 levels.
If $p < 1/4$ or in this case c or $d > 1/4$, then $a+b+c+d > 1$ leads to impossible scenario.
So, no codeword of length 1 can be formed.
Contrapositively, if there is codeword for p of length 1, then c and d must be both $\leq 1/4$.
Considering case where if $c > 1/4$, then $a+b+c+d > 1$, which makes it impossible to construct Huffman code.
Hence by contradiction, if $p < 1/4$, then the corresponding symbol will have codeword of length 1.

6. Shannon Code : $l_i = \lceil \log 1/p_i \rceil$

a) (0.5 , 0.25 , 0.125 , 0.125)

(Codeword)

Symbol	Probability	F_i (Decimal)	F_i (Binary)	l_i	C_i
1	0.5	0.0	0.00	1	0
2	0.25	0.5	0.10	2	10
3	0.125	0.75	0.110	3	110
4	0.125	0.875	0.111	3	111

b) Show avg length satisfies $H(x) \leq L < H(x) + 1$ as well as prefix tree.

$l_i = \lceil \log \frac{1}{p_i} \rceil \rightarrow \log \frac{1}{p_i} \leq l_i < \log \frac{1}{p_i} + 1$

Above means that $H(x) \leq L = \sum p_i l_i < H(x) + 1$.

(Hence Proved)

Now, by choice of l_i, we have -

$2^{-l_i} \leq p_i < 2^{-(l_i - 1)}$

Thus F_j where $j > i$ is different from F_i by atleast 2^{-l_i} and will differ from F_i by atleast one place in first l_i bits of binary representation of F_i.

Hence the codeword for F_j, $j > i$, which has length $l_j \geq l_i$ is different from codeword for F_i by at least one in the first l_i places.

Thus, no codeword is prefix of any other codeword.

(Hence Proved)

In other words, first l_i characters of F_j must be different from F_i.

So, we can write this as -

$$F_j - F_i > 2^{-l_i}$$

$$LHS = F_j - F_i$$

$$= p_i + p_{i+1} + \cdots + p_{j-1} + p_j$$

$$RHS = 2^{-l_i}$$

$$= 2^{-\log_2(1/p_i)}$$

$$= 2^{\log_2(p_i)}$$

$\Rightarrow F_j - F_i > 2^{-l_i}$ is always true.