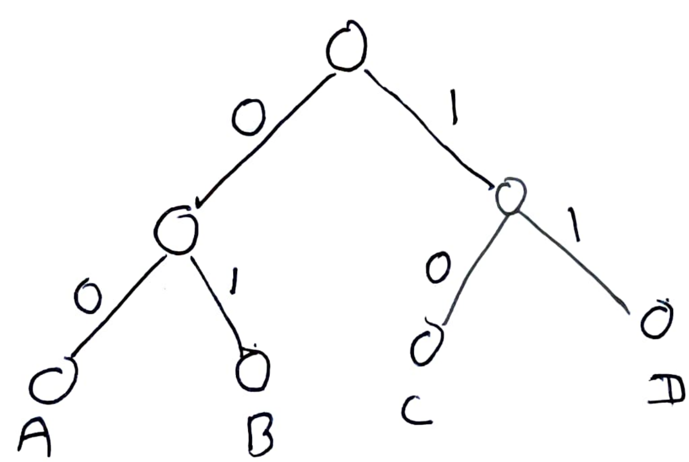
Codes as Tree

Fixed length code.

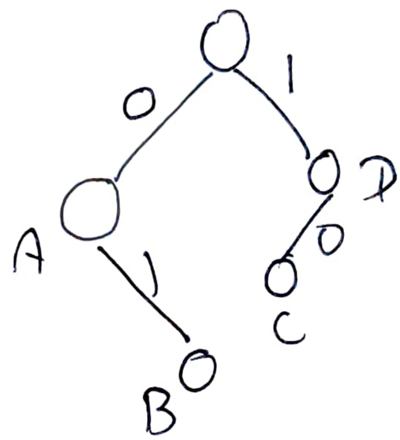
A	00
B	01
C	10
D	11



No issue in terms of fuzzyness. But not good for data compression.

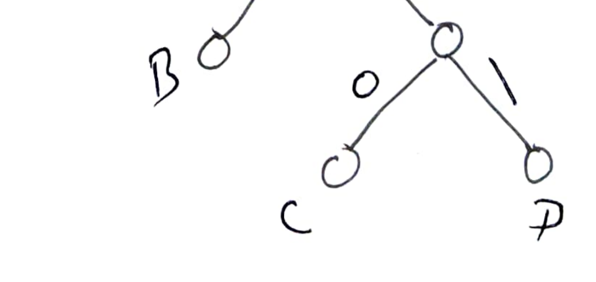
V. L. E (non-prefix-free)

A	0
B	01
C	10
D	1



V. L. E (prefix-free) — "ABD"
Traverse for

A	0
B	10
C	110
D	111



The first and third trees causes polarity to the two prefix free code and share one common property:- only leaves are labeled.

However in second tree!. two non-leaves are labeled.

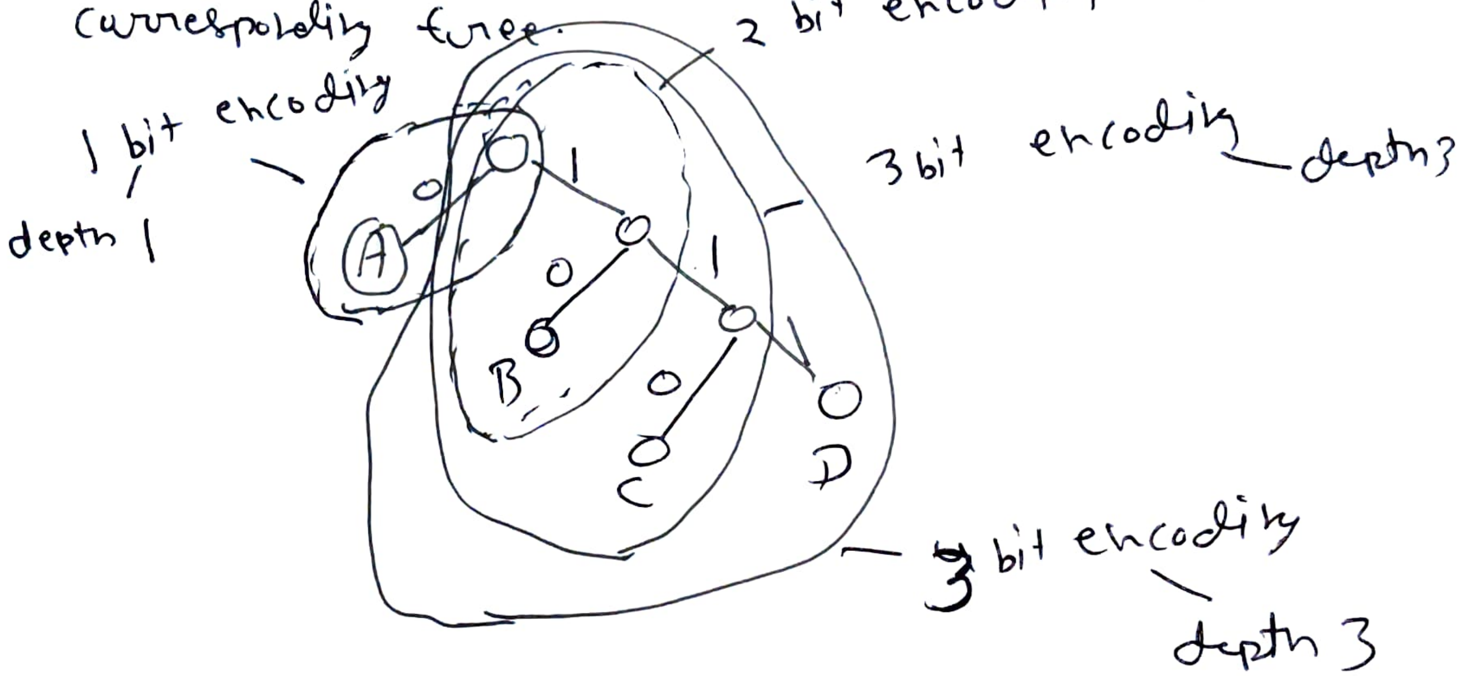
The encoding of a symbol "a" is a prefix of that of another symbol "b" iff the node labled a is an ancestor of the node labled 'b.

So how we can define Σ-tree which is a binary tree with leaves labled in one-to-one correspondence with the symbol of Σ.

For a Σ-tree "T" and symbol frequencies $p = \{p_a\}_{a \in \Sigma}$, we denote by $L(T, p)$ the average depth of a leaf in T with the contribution of each leaf weighted according to the frequency of its label.

$$L(T, p) = \sum_{a \in \Sigma} p_a \cdot (\text{depth of the leaf labled } a \text{ in } T)$$

(Encoding length and tree depth) :- For every binary code, the encoding length in bits of a symbol $a \in \Sigma$ equals the depth of the node with label a in the corresponding tree.



Rephrased Problem :-

> Input :- A non-negative frequency p_a for each symbol a of an alphabet Σ of size $n \geq 2$
>
> output :- A Σ - tree with minimum possible average leaf depth