Yes

Ambiguity

10

3　　21

Comma is an extra alphabet

{0, 1}

{0, 1, ","}

ternary

10,1,1,1,0,1,0,1,1,

Morse code

$\left.\begin{array}{l}\text{dits}\\ \text{dashes}\\ \text{letter space}\\ \text{word space}\end{array}\right\}$ 4 alphabets

Def: A source code C for a random variable X is a mapping from $\mathcal{X}$, the range of X, to $\{0,1\}^*$, the set of all finite-length strings of symbols from $\{0,1\}$.

Let $C(x)$ denote the codeword corresponding to x and let $l(x)$ denote the length of $C(x)$.

$C(3) = 110$

$l(3) = 3$

Defn: The expected length $L(C)$ of a source code $C(x)$ for a random variable X is given by

$$L(C) = \sum_{x} p(x)\, l(x)$$

Defn: Non-singular code

A code is said to be non-singular if every element of the range of X maps into a different bit string.

i.e.

$$x \neq x' \Rightarrow C(x) \neq C(x')$$

X	C
1	10
2	11
3	10

singularity.

Don't want this.

Defn: The extension C^* of a code C is the mapping from finite-length strings of X to finite-length bit-strings, defined by

$$C(x_1 x_2 x_3 \cdots x_n)$$
$$= C(x_1) C(x_2) \cdots C(x_n),$$

where the RHS denotes the concatenation of $C(x_1), C(x_2), \ldots, C(x_n)$

$C(1232) =$

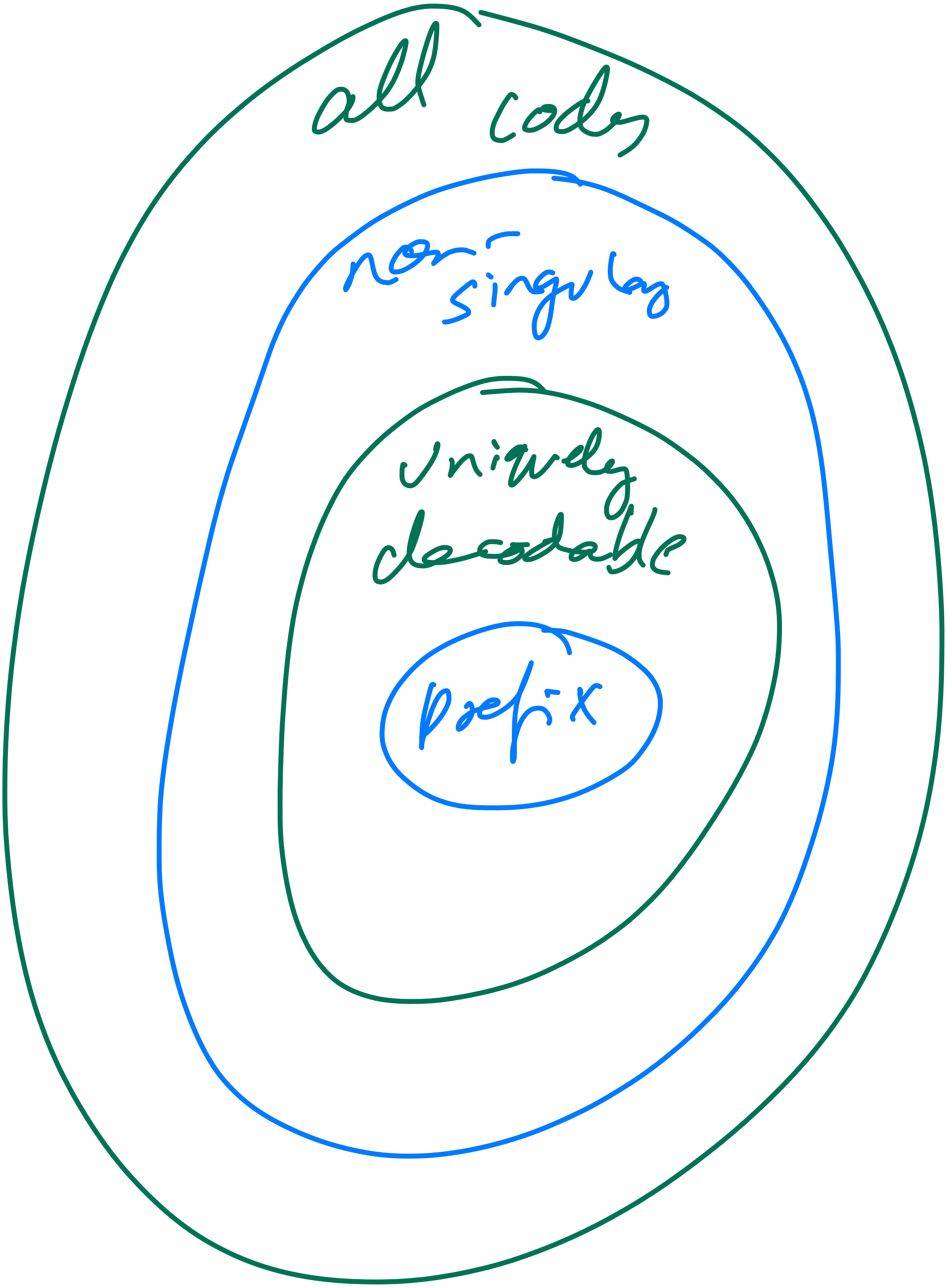
$C(1)\ C(2)\ C(3)\ C(1)\ C(2)$

$= 0\,10\,110\,0\,10$

Defn: A code is called uniquely decodable if its extension is non-singular.

Defn: A code is called a prefix code or an instantaneous code if no codeword is a prefix of any other codeword.

Classes of codes



X	A	B	C	D
1	0	0	10	0
2	1	010	00	10
3	0	01	11	110
4	1	10	110	111
	Singular	Singular ✗ Extension is singular not U.D.	non-singular not prefix U.D.	Prefix

110100111

110, 10, 0, 111

don't need memory

010 → 2, 14

110

100000

x 11 0000 00 0000 11 →

3 2 2 2 2 2 3

Theorem: Kraft inequality

For any prefix binary code, the codeword lengths $l_1, l_2, \ldots, l_m$ must satisfy

$$\sum_i 2^{-l_i} \leq 1$$

e.g. 2, 2, 2, 3, 3, 3, 3

$$2^{-2} + 2^{-2} + 2^{-2} + 2^{-3} \times 4$$

$$= \frac{3}{4} + \frac{1}{8} \times 4 = \frac{5}{4} > 1$$

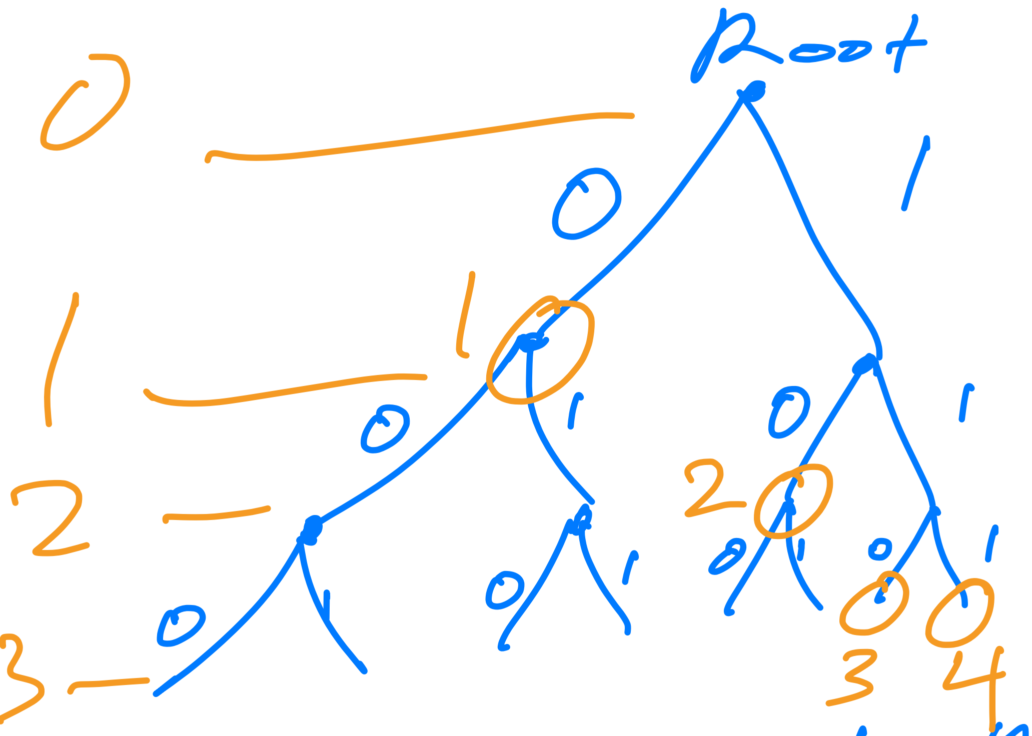
not possible

$$2^{-1}+2^{-1}+2^{-2}+2^{-2}$$

$$=\frac{1}{2}+\frac{1}{2}+\frac{1}{4}+\frac{1}{4}=\frac{3}{2}>1$$

not possible

Proof: Consider a binary tree. Each node has 2 children. The branches of the tree are 0 and 1. Each codeword is represented by a leaf on the tree



0
10
110
111

Codeword is path from the root to the leaf.

Prefix code: no codeword is an ancestor of any other codeword. Each codeword eliminates its descendants as possible codewords

Let l_{max} be the length of the longest codeword. e.g. 3.

Consider all the nodes at level l_{max}.

Some of them are codewords, some of them are descendants of codewords some are neither.

A codeword at level l_i has

$2^{l_{max} - l_i}$ descendants

at level l_{max}.

All of them are disjoint.

$$\sum 2^{l_{max} - l_i} \leq 2^{l_{max}}$$

summation of all possible descendants at level l_{max}

no. of nodes at level l_{max}

$2^2 + 2^1 + 2^0 + 2^0 \leq 2^3$

0 10 110 111

$8 \leq 8$

$$\sum 2^{-l_i} \leq 1$$

Kraft inequality

Conversely, given a set of codeword lengths that satisfy the inequality $\exists$ an instantaneous / prefix code with these word lengths.

Construct the tree.
label the first node
of depth l_1 as
codeword 1 and remove
all its descendants
from the tree. Then
label the first
remaining node of
depth l_2 as codeword 2
and delete all its descendants
and so on.

Optimal codes

Minimize

$$L = \sum p_i l_i$$

over all integers

$l_1, l_2, \ldots, l_m$

satisfying

$$\sum 2^{-l_i} \leq 1$$

Assumption: neglect the integer constraint on l_i.

Assume equality on the constraint:

$$\sum 2^{-l_i} = 1$$

Lagrange multiplier

$$J = \sum p_i l_i + \lambda \sum 2^{-l_i}$$

$$\frac{\partial J}{\partial l_i} = p_i - \lambda 2^{-l_i} \log_e 2$$

$$2^{-l_i} = \frac{p_i}{\lambda \log_e 2}$$

$$\lambda = \frac{1}{\log_e 2}$$

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

$$\Rightarrow \boxed{l_i^* = -\log_2 p_i}$$

$$L^* = \sum p_i \, l_i^*$$

$$= -\sum p_i \log_2 p_i$$

$$= H(X)$$