Theory:-

Entropy:-

- The "Entropy" is defined as the average information per message. It is denoted by H and its units are bits/messages.
- The entropy must be as high as possible in order to ensure maximum transfer of information. We will prove that the entropy depends only on the probabilities of the symbols in the alphabet of the source.

$$\boxed{H=I1*p1+I2*p2+I3*p3+I4*p4}$$

Information Rate (R): -

If the source of the messages generates number of messages per second then theinformation rate is given as,

$$\boxed{R=r x H}$$

Where, r= Number of messages/secs, and H= Average information/message.

Units of information rate:

R =[r(messages/sec) *H(information /message)]

R = Average information per second expressed in bits/sec.

exp 1

AIM:- Implementation of Huffman Code using SCILAB.

Theory:
Huffman code:

Entropy:-
- The "Entropy" is defined as the average information per message. It is denoted by H and its units are bits/messages.
- The entropy must be as high as possible in order to ensure maximum transfer of information. We will prove that the entropy depends only on the probabilities of the symbols in the alphabet of the source.

$$\boxed{H=I1*P1+I2*P2+I3*P3+I4*P4}$$

Length of code (L) :-

$$\boxed{L=L1*P1+L2*P2+L3*P3+L4*P4}$$

Efficiency :
The efficiency of a coding system is the ratio of the average information per symbol to the average code length.

$$\boxed{E=H/L *100 \ \%}$$

Redundancy :

$$\boxed{R=100-E \ \%}$$

exp 2

Problems:

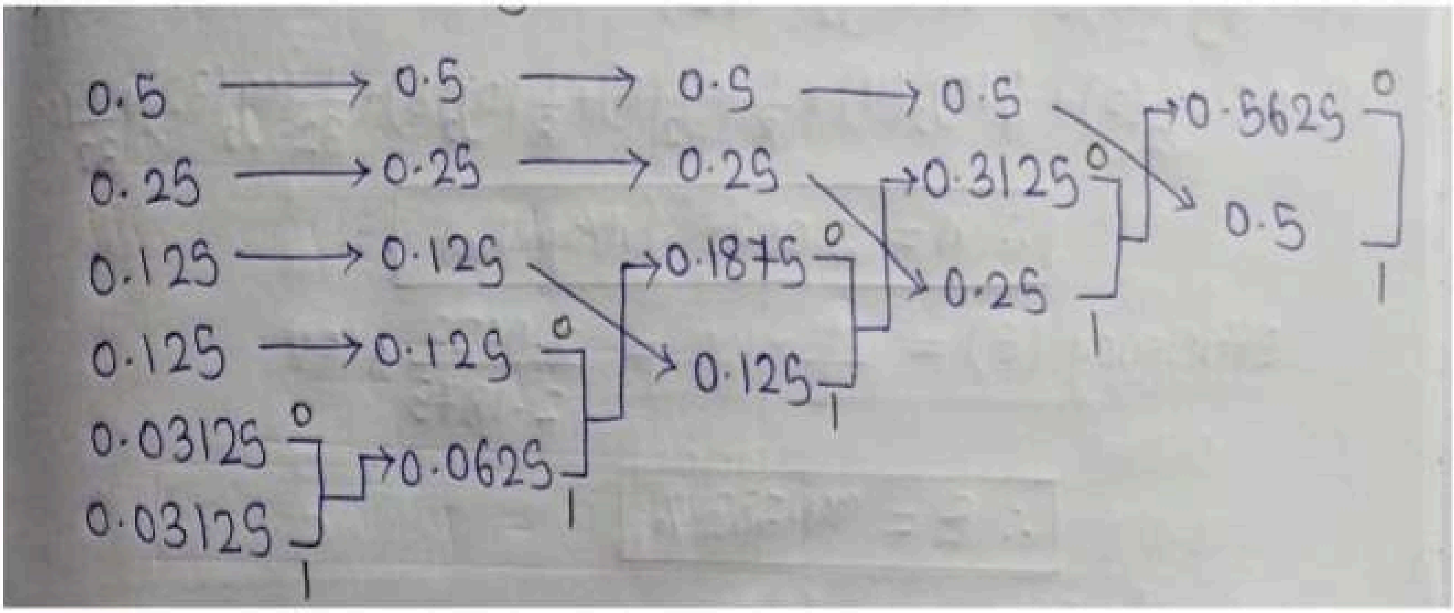
Find Entropy, codeword length, efficiency and redundancy by using HuffmanCoding of following data given :

Q1).

Messages	m1	m2	m3	m4	m5	m6
Probabilities	1/2	1/4	1/8	1/8	1/32	1/32

Solution:

Huffman Coding



Huffman Coding table

Messages	Probabilities	Codeword	Codeword length
m1	1/2	1	1
m2	1/4	10	2
m3	1/8	100	3
m4	1/8	0000	4
m5	1/32	01000	5

exp 2 ex

Aim: Implementation of Linear Block Code using Scilab.

exp 4

Software: Scilab open source.

Theory:

Linear Block Code:

In the linear block codes, the parity bits and message bits have a linear combination, which means that the resultant code word is the linear combination of any two code words.

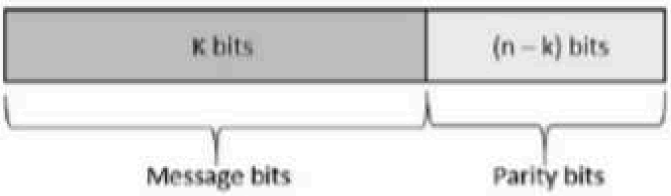
Let us consider some blocks of data, which contains **k** bits in each block. These bits are mapped with the blocks which has **n** bits in each block. Here **n** is greater than **k**. The transmitter adds redundant bits which are n−kn−k bits. The ratio **k/n** is the **code rate**. It is denoted by **r** and the value of **r** is **r < 1**.

The n−kn−k bits added here, are **parity bits**. Parity bits help in error detection and error correction, and also in locating the data. In the data being transmitted, the left most bits of the code word correspond to the message bits, and the right most bits of the code word correspond to the parity bits.

Any linear block code can be a systematic code, until it is altered. Hence, an unaltered block code is called as a **systematic code**.

Following is the representation of the **structure of code word**, according to their allocation.

Structure of code word



Generator matrix:a generator matrix is a matrix whose rows form a basis for a linear code. The codewords are all of the linear combinations of the rows of this matrix, that is, the linear code is the row space of its generator matrix.

$\boxed{G=[Ik : Pk]}$

Parity check matrix :, a parity check matrix, H of a linear code C is a generator matrix of the dual code, C⊥. This means that a codeword c is in C if and only if the

matrix-vector product $Hc^T = 0$ (some authors[1] would write this in an equivalent form, $cH^T = 0$.)

The rows of a parity check matrix are the coefficients of the parity check equations.[2] That is, they show how linear combinations of certain digits (compc of each codeword equal zero. The parity check matrix is

$\boxed{H=[P^T : Ik]}$