Illustration I :

Let us consider a binary source with source alphabet $S = \{s_1, s_2\}$ with probabilities

$$P = \left\{\frac{1}{256}, \frac{255}{256}\right\}$$

$$\text{Then, entropy } H(S) = \sum_{i=1}^{2} p_i \log \frac{1}{p_i}$$

$$= \frac{1}{256} \log 256 + \frac{255}{256} \log \frac{256}{255}$$

$$= \mathbf{0.037 \text{ bits/message symbol}}$$

$\therefore$ The average uncertainty is very very small and is relatively very easy to guess whether s_1 or s_2 will occur.

Illustration II :

$$\text{Let } S' = \{s_3, s_4\} \text{ with } P' = \left\{\frac{7}{16}, \frac{9}{16}\right\}$$

$$\text{Then, entropy } H(S') = \frac{7}{16} \log \frac{16}{7} + \frac{9}{16} \log \frac{16}{9}$$

$$= \mathbf{0.989 \text{ bits/message symbol}}$$

In this case, it is hard to guess whether s_3 or s_4 is transmitted.

Illustration III :

$$\text{Let } S'' = \{s_5, s_6\} \text{ with } P'' = \left\{\frac{1}{2}, \frac{1}{2}\right\}$$

$$\text{Then, entropy } H(S'') = \frac{1}{2} \log 2 + \frac{1}{2} \log 2 = 1 \text{ bit/message symbol}$$

In this case, the uncertainty is maximum for a binary source and becomes impossible to guess which symbol is transmitted.

These illustrations clearly indicate the significance and dependence of entropy on probabilities of messages.

INFORMATION RATE : Let us suppose that the symbols are emitted by the source at a fixed time rate "r_s" symbols/sec. The ***"average source information rate R_s"*** in bits/sec is defined as the product of the average information content per symbol and the message symbol rate r_s.

$$\therefore \quad R_s = r_s H(S) \text{ bits/sec or BPS} \qquad \text{...... (1.5)}$$

Example 2.17 : Given the messages x_1, x_2, x_3, x_4, x_5 and x_6 with respective probabilities 0.4, 0.2, 0.2, 0.1, 0.07 and 0.03, construct a binary code by applying Shannon-Fano encoding procedure. Determine code efficiency and redundancy of the code.

Solution

There are two ways in which Step No. 2 given in the procedure can be applied. Let us discuss both the ways.

1st Way :

	p_i								Code	l_i in binits
x_1	0.4	1	0.4	1					11	2
x_2	0.2	1	0.2	0					10	2
x_3	0.2	0	0.2	1					01	2
x_4	0.1	0	0.1	0	0.1	1			001	3
x_5	0.07	0	0.07	0	0.07	0	007	1	0001	4
x_6	0.03	0	0.03	0	0.03	0	003	0	0000	4

Table 2.24 : Code-table for example 2.17

$$\text{Average length } L = \sum_{i=1}^{6} p_i l_i$$

$$= (0.4)(2) + (0.2)(2) + (0.2)(2) + (0.1)(3) + (0.07)(4) + (0.03)(4)$$

$$= 2.3 \text{ binits/message-symbol}$$

Entropy

$$H(S) = \sum_{i=1}^{6} p_i \log \frac{1}{p_i}$$

$$= 0.4 \log \frac{1}{0.4} + 2 \times 0.2 \log \frac{1}{0.2} + 0.1 \log \frac{1}{0.1} + 0.07 \log \frac{1}{0.07} + 0.03 \log \frac{1}{0.03}$$

$$= 2.209 \text{ bits/message-symbol}$$

$\therefore$ Code efficiency $\eta_c = \frac{H(S)}{L} = \frac{2.209}{2.3}$

$= 96.04\%$

$\therefore$ Code redundancy $R_{\eta c} = 3.96\%$

2nd Way :

	P_i								Code	l_i in binits
x_1	0.4	1							1	1
x_2	0.2	0	0.2	1	0.2	1			011	3
x_3	0.2	0	0.2	1	0.2	0			010	3
x_4	0.1	0	0.1	0	0.1	1			001	3
x_5	0.07	0	0.07	0	0.07	0	0.07	1	0001	4
x_6	0.03	0	0.03	0	0.03	0	0.03	0	0000	4

Table 2.25 : Code-table for example 2.17

Observation of tables 2.24 and 2.25 reveals that both are instantaneous codes with some change in the coding pattern.

$$\text{Average length, } L = \sum_{i=1}^{6} p_i l_i$$

$$= (0.4)(1) + (0.2)(3) + (0.2)(3) + (0.1)(3) + (0.07)(4) + (0.03)(4)$$

$= 2.3$ binits/message-symbol

Entropy, $H(S) = 2.209$ bits/message-symbol

$\therefore$ Code efficiency $\eta_c = \frac{H(S)}{L} = \frac{2.209}{2.3}$

$= 96.04\%$ which remains the same for both ways of coding.

The code-trees corresponding to both the ways can be drawn as shown in figure 2.5 (a) and (b).

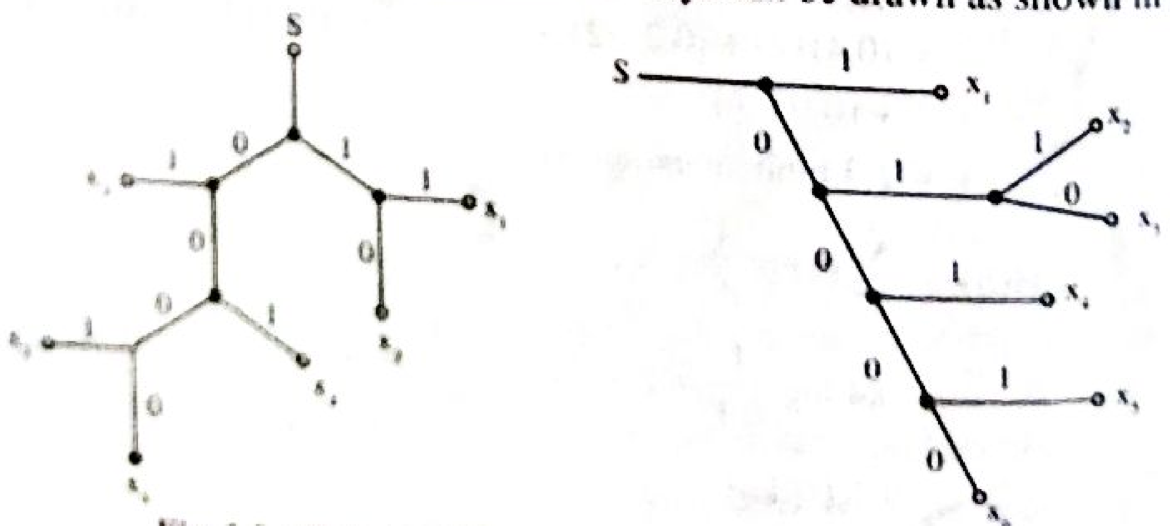

Fig. 2.5 : Code-tree for (a) 1st way (b) 2nd way of example 2.17.

Example 2.18 : You are given 4 messages x_1, x_2, x_3 and x_4 with respective probabilities 0.1, 0.2, 0.3, 0.4.

(i) Device a code with prefix property (Shannon-Fano code) for these messages and draw the code tree.

(ii) Calculate the efficiency and redundancy of the code.

(iii) Calculate the probabilities of 0's and 1's in the code.

Solution

(i)

	p_i						Code	l_i in binits
x_4	0.4	1					1	1
x_3	0.3	0	0.3	1			01	2
x_2	0.2	0	0.2	0	0.2	1	001	3
x_1	0.1	0	0.1	0	0.1	0	000	3

Table 2.26 : Code-table for example 2.18.

(ii) Average length, $L = \sum_{i=1}^{4} p_i l_i$

$$= (0.4)(1) + (0.3)(2) + (0.2)(3) + (0.1)(3)$$

$$= 1.9 \text{ binits/message-symbol}$$

$$\text{Entropy } H(S) = \sum_{i=1}^{4} p_i \log \frac{1}{p_i}$$

$$= 0.4 \log \frac{1}{0.4} + 0.3 \log \frac{1}{0.3} + 0.2 \log \frac{1}{0.2} + 0.1 \log \frac{1}{0.1}$$

$$= 1.846 \text{ bits/message-symbol}$$

$$\therefore \text{Code efficiency } \eta_c = \frac{H(S)}{L} = \frac{1.846}{1.9} = 97.15\%$$

$$\therefore \text{Code redundancy } R_{\eta_c} = 2.85\%$$

The code-tree can be drawn as shown in figure 2.6.

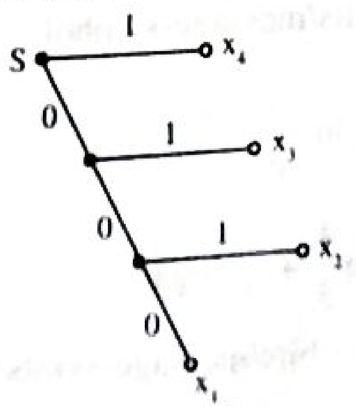

Fig. 2.6 : Code-tree for example 2.18

(iii) The probability of '0's and '1's in the code are found using the formulas,

$$P(0) = \frac{1}{L}\sum_{i=1}^{4}[\text{Number of '0's in the code for } x_i]\,[p_i] \quad \text{......(2.35)}$$

$$P(1) = \frac{1}{L}\sum_{i=1}^{4}[\text{Number of '1's in the code for } x_i]\,[p_i] \quad \text{......(2.36)}$$

From the code-table 2.26, we have

$$P(0) = \frac{1}{1.9}[(3)(0.1) + (2)(0.2) + (1)(0.3) + (0)(0.4)]$$

$$\therefore \quad \mathbf{P(0) = 0.5263}$$

$$\text{and} \quad P(1) = \frac{1}{1.9}[(0)(0.1) + (1)(0.2) + (1)(0.3) + (1)(0.4)]$$

$$\therefore \quad \mathbf{P(1) = 0.4737}$$

Example 2.19 : Consider a source $S = \{s_1, s_2\}$ with probabilities 3/4 and 1/4 respectively. Obtain Shannon-Fano code for source S, its 2^{nd} and 3^{rd} extensions. Calculate efficiencies for each case.

Solution

For the basic source:

	p_i
s_1	3/4
s_2	1/4

Code	length l_i in binits
1	1
0	1

Table 2.27 : Code-table for basic source of example 2.19

$$\text{Average length, } L = \sum_{i=1}^{2} p_i l_i$$

$$= \frac{3}{4}(1) + \frac{1}{4}(1)$$

$$= 1 \text{ binits/message-symbol}$$

$$\text{Entropy, } H(S) = \sum_{i=1}^{2} p_i \log\frac{1}{p_i}$$

$$= \frac{3}{4}\log\frac{4}{3} + \frac{1}{4}\log 4$$

$$= 0.8113 \text{ bits/message-symbol}$$

$$\therefore \text{ Code efficiency } \eta_c = \eta_c^{(1)} = \frac{H(S)}{L} = \frac{0.8113}{1} = 81.13\%$$

$$\therefore \quad \eta_c^{(1)} = \mathbf{81.13\%}$$

For the 2nd extension : The 2nd extension will have $2^2 = 4$ symbols given by s_1s_1, s_1s_2, s_2s_1 and s_2s_2 with probabilities 9/16, 3/16, 3/16 and 1/16 respectively.

	P_i						Code	l_i in binits
s_1s_1	9/16	1					1	1
s_1s_2	3/16	0	3/16	1			01	2
s_2s_1	3/16	0	3/16	0	3/16	1	001	3
s_2s_2	1/16	0	1/16	0	1/16	0	000	3

Table 2.28 : Code-table for 2nd extension of example 2.19

The average length L_2 of the 2nd extension is given by

$$L_2 = \sum_{i=1}^{4} p_i l_i$$

$$= \left(\frac{9}{16}\right)(1) + \left(\frac{3}{16}\right)(2) + \left(\frac{3}{16}\right)(3) + \left(\frac{1}{16}\right)(3)$$

$$= 1.6875 \text{ bits/message-symbol}$$

Entropy of the 2nd extended source is given by equation (1.33) as

$$H(S^2) = 2\,H(S)$$

$$= 2\,[0.8113]$$

$$= 1.6226 \text{ bits/message-symbol}$$

$\therefore$ Code efficiency of the 2nd extended source is

$$\eta_c^{(2)} = \frac{H(S^2)}{L_2} = \frac{1.6226}{1.6875} = 96.15\%$$

$$\therefore \quad \eta_c^{(2)} = \mathbf{96.15\%}$$

3rd extension : The 3rd extension will have $2^3 = 8$ symbols which are listed below in the non increasing order.

Symbol	p_i								Code	l_i in binits
$s_1s_1s_1$	27/64	1	27/64	1					11	2
$s_1s_1s_2$	9/64	1	9/64	0					10	2
$s_1s_2s_1$	9/64	0	9/64	1	9/64	1			011	3
$s_2s_1s_1$	9/64	0	9/64	1	9/64	0			010	3
$s_1s_2s_2$	3/64	0	3/64	0	3/64	1	3/64	1	0011	4
$s_2s_1s_2$	3/64	0	3/64	0	3/64	1	3/64	0	0010	4
$s_2s_2s_1$	3/64	0	3/64	0	3/64	0	3/64	1	0001	4
$s_2s_2s_2$	1/64	0	1/64	0	1/64	0	1/64	0	0000	4

Table 2.29 : Code-table for 3rd extension of example 2.19

The average length L_3 of the 3rd extended source is

$$L_3 = \sum_{i=1}^{8} p_i l_i$$

$$= \left(\frac{27}{64}\right)(2) + \left(\frac{9}{64}\right)(2) + \left(\frac{9}{64}\right)(3) + \left(\frac{9}{64}\right)(3) + \left(\frac{3}{64}\right)(4)$$

$$+ \left(\frac{3}{64}\right)(4) + \left(\frac{3}{64}\right)(4) + \left(\frac{1}{64}\right)(4)$$

$$= 2.59375 \text{ binits/message-symbols}$$

The entropy of the 3rd extended source is given by equation (1.34) as,

$$H(S^3) = 3\,H(S)$$

$$= 3\,[0.8113] = 2.4339 \text{ bits/message-symbol}$$

∴ Code efficiency of the 3rd extended source is

$$\eta_c^{(3)} = \frac{H(S^3)}{L_3} = \frac{2.4339}{2.59375} = 93.84\%$$

$$\therefore \quad \eta_c^{(3)} = 93.84\%$$

Which is less than the efficiency of the 2nd extended source indeed!!!; And Shannon's first theorem of equation (2.33) is violated! The following reasoning appears to be relevant for the above case.

While following step no. 2 of the Shannon-Fano procedure for 3rd extension, we had grouped ($s_1s_1s_1$) and ($s_1s_1s_2$) with a total probability of (36/64) in one group and the remaining symbols with a total probability of (28/64) in the other group. We could have grouped $s_1s_1s_1$ with probability (27/64) in one group and the rest of the symbols with a total probability of (37/64) in the other group. With the former groupings, the difference in total probability is (36/64 – 28/64 = 8/64) which is less than the difference in total probability of (37/64 – 27/64 = 10/64) for the latter case. According to the rule formulated by Fano, the least difference case has to be considered and in doing so, we get a decreasing efficiency for the 3rd extension case. If we violate this rule and consider the latter case, we can achieve increasing efficiency as shown below:

	P_i												Code	l_i in binits
$s_1s_1s_1$	27/64	1											1	1
$s_1s_1s_2$	9/64	0	9/64	1	9/64	1							011	3
$s_1s_2s_1$	9/64	0	9/64	1	9/64	0							010	3
$s_2s_1s_1$	9/64	0	9/64	0	9/64	1							001	3
$s_1s_2s_2$	3/64	0	3/64	0	3/64	0	3/64	1					0001	4
$s_2s_1s_2$	3/64	0	3/64	0	3/64	0	3/64	0	3/64	1			00001	5
$s_2s_2s_1$	3/64	0	3/64	0	3/64	0	3/64	0	3/64	0	3/64	1	000001	6
$s_2s_2s_2$	1/64	0	1/64	0	1/64	0	1/64	0	1/64	0	1/64	0	000000	6

Table 2.30 : Code-table for 3rd extension of example 2.19.

The average length L_3 referring to table 2.30 is given by

$$L_3 = \sum_{i=1}^{8} p_i l_i$$

$$= \left(\frac{27}{64}\right)(1) + \left(\frac{9}{64}\right)(3) + \left(\frac{9}{64}\right)(3) + \left(\frac{9}{64}\right)(3) + \left(\frac{3}{64}\right)(4)$$

$$+ \left(\frac{3}{64}\right)(5) + \left(\frac{3}{64}\right)(6) + \left(\frac{1}{64}\right)(6)$$

$$L_3 = 2.484375 \text{ binits/message-symbol}$$

Entropy $H(S^3) = 3\,H(S)$

$$= 2.4339 \text{ bits/message-symbol}$$

$\therefore$ Code efficiency of the 3rd extended source is

$$\eta_c^{(3)} = \frac{H(S^3)}{L_3} = \frac{2.4339}{2.484375}$$

$$= 97.97\%$$

Which is greater than the second-extension efficiency of 96.15%.

Note : From the above, we can conclude that the symbol with highest probability should be made to correspond to a code with a shortest word-length.

Example 2.20 : Consider a discrete memoryless source whose alphabet consists of K equiprobable symbols.

Example **2.26 :** Given the messages x_1, x_2, x_3, x_4, x_5 and x_6 with respective of probabilities of 0.4, 0.2, 0.2, 0.1, 0.07 and 0.03, construct a binary code by applying Huffman encoding procedure. Determine the efficiency and redundancy of the code so formed.

Solution

Source Symbols	p_i	Code	Source S_a		Source S_b		Source S_c		Source S_d	
			p_i	Code	p_i	Code	p_i	Code	p_i	Code
x_1	0.4	1	0.4	1	0.4	1	0.4	1	0.6	0
x_2	0.2	01	0.2	01	0.2	01	0.4	00	0.4	1
x_3	0.2	000	0.2	000	0.2	000	0.2	01		
x_4	0.1	0010	0.1	0010	0.2	001				
x_5	0.07	00110	0.1	0011						
x_6	0.03	00111								

Table 2.44 : Huffman code-table for example 2.26

The average length L is given by

$$L = \sum_{i=1}^{6} p_i l_i$$

$$= (0.4)(1) + (0.2)(2) + (0.2)(3) + (0.1)(4) + (0.07)(5) + (0.03)(5)$$

$$= 2.3 \text{ binits/message-symbol}$$

The entropy H(S) is given by

$$H(S) = \sum_{i=1}^{6} p_i \log \frac{1}{p_i}$$

$$= 0.4 \log \frac{1}{0.4} + 2 \times 0.2 \log \frac{1}{0.2} + 0.1 \log \frac{1}{0.1} + 0.07 \log \frac{1}{0.07} + 0.03 \log \frac{1}{0.03}$$

$$= 2.209 \text{ bits/message-symbol}$$

$$\therefore \text{Code efficiency } \eta_c = \frac{H(S)}{L} = \frac{2.209}{2.3}$$

$$= 96.04\%$$

$$\therefore \text{Code redundancy } R_{\eta c} = 2.96\%$$

Example 2.27 : Consider a zero-memory source with

$$S = \{s_1, s_2, s_3, s_4, s_5, s_6, s_7\}$$

$$P = \{0.4, 0.2, 0.1, 0.1, 0.1, 0.05, 0.05\}$$

(i) Construct a binary Huffman code by placing the composite symbol as low as you can.

(ii) Repeat (i) by moving the composite symbol ***"as high as possible"***.

In each of the cases (i) and (ii) above, compute the variances of the word-lengths and comment on the result.

Solution

For Huffman binary coding, dummy symbols are not required.

(i) Table 2.45 below gives the complete code-table with the composite symbol placed "as low as possible".

Source Symbols	p_i	Code	Source S_a		Source S_b		Source S_c		Source S_d		Source S_e	
			p_i	Code	p_i	Code	p_i	Code	p_i	Code	p_i	Code
s_1	0.4	1	0.4	1	0.4	1	0.4	1	0.4	1	0.6	0
s_2	0.2	01	0.2	01	0.2	01	0.2	01	0.4	00	0.4	1
s_3	0.1	0010	0.1	0010	0.2	000	0.2	000	0.2	01		
s_4	0.1	0011	0.1	0011	0.1	0010	0.2	001				
s_5	0.1	0000	0.1	0000	0.1	0011						
s_6	0.05	00010	0.1	0001								
s_7	0.05	00011										

Table 2.45 : Code-table for example 2.27 with composite symbol placed "as low as possible"

The average length L is given by,

$$L = \sum_{i=1}^{7} p_i l_i$$

$$= (0.4)(1) + (0.2)(2) + (0.1)(4) + (0.1)(4) + (0.1)(4) + (0.05)(5) + (0.05)(5)$$

$$= 2.5 \text{ binits/message-symbol}$$

From the equation (0.37), the variance is defined as

$$\text{Var}(X) = E[(X - \mu)^2]$$

where μ = average value.

The variance of word-lengths is calculated from

$$\text{Var}(l_i) = E[(l_i - L)^2] \qquad \text{...... (2.38)}$$

$$= \sum_{i=1}^{7} p_i (l_i - L)^2$$

$$= (0.4)(1 - 2.5)^2 + (0.2)(2 - 2.5)^2 + (0.1)(4 - 2.5)^2 + (0.1)(4 - 2.5)^2 + (0.1)(4 - 2.5)^2 + (0.05)(5 - 2.5)^2 + (0.05)(5 - 2.5)^2$$

$$= 2.25$$

(ii) Table 2.46 gives the complete code-table with the composite symbol placed ***"as high as possible"***.

Source Symbols	p_i	Code	Source S_a		Source S_b		Source S_c		Source S_d		Source S_e	
			p_i	Code	p_i	Code	p_i	Code	p_i	Code	p_i	Code
s_1	0.4	00	0.4	00	0.4	00	0.4	00	0.4	1	0.6	0
s_2	0.2	11	0.2	11	0.2	10	0.2	01	0.4	00	0.4	1
s_3	0.1	011	0.1	010	0.2	11	0.2	10	0.2	01		
s_4	0.1	100	0.1	011	0.1	010	0.2	11				
s_5	0.1	101	0.1	100	0.1	011						
s_6	0.05	0100	0.1	101								
s_7	0.05	0101										

Table 2.46 : Code-table for example 2.27 with composite symbol placed "as high as possible".

The average length L is given by

$$L = \sum_{i=1}^{7} p_i l_i$$

$$= (0.4)(2) + (0.2)(2) + (0.1)(3) + (0.1)(3) + (0.1)(3) + (0.05)(4) + (0.05)(4)$$

$$= 2.5 \text{ binits/message-symbol}$$

From equation (2.38), the variance of word-lengths is

$$\text{Var}(l_i) = E[(l_i - L)^2]$$

$$= \sum_{i=1}^{7} p_i (l_i - L)^2$$

$$= (0.4)(2-2.5)^2 + (0.2)(2-2.5)^2 + (0.1)(3-2.5)^2 + (0.1)(3-2.5)^2 + (0.1)(3-2.5)^2 + (0.05)(4-2.5)^2 + (0.05)(4-2.5)^2$$

$$= 0.45$$

Comment : When the composite symbol is moved as high as possible, the variance of the word-lengths over the ensemble of source symbols would become smaller, which, indeed is desirable.

Example 2.28 : Consider a source with 8 alphabets A to H with respective probabilities of 0.22, 0.20, 0.18, 0.15, 0.10, 0.08, 0.05, 0.02.

(a) Construct a binary compact (Huffman) code and determine the code efficiency.

(b) Construct a ternary compact code and determine the efficiency of the code.

(c) Construct a quarternary compact code and determine the code efficiency. Compare and comment on the result. Draw code-trees for all three cases.

(a) Binary Code :

Source Symbols	p_i	Code	Source S_a		Source S_b		Source S_c		Source S_d		Source S_e		Source S_f	
			p_i	Code	p_i	Code	p_i	Code	p_i	Code	p_i	Code	p_i	Code
A	0.22	10	0.22	10	0.22	10	0.25	01	0.33	00	0.42	1	0.58	0
B	0.20	11	0.20	11	0.20	11	0.22	10	0.25	01	0.33	00	0.42	1
C	0.18	000	0.18	000	0.18	000	0.20	11	0.22	10	0.25	01		
D	0.15	001	0.15	001	0.15	001	0.18	000	0.20	11				
E	0.10	011	0.10	011	0.15	010	0.15	001						
F	0.08	0100	0.08	0100	0.10	011								
G	0.05	01010	0.07	0101										
H	0.02	01011												

Table 2.47 : Code-table for example 2.28 (a)

Solution :

The average length $L^{(2)}$ given by

$$L^{(2)} = \sum_{i=1}^{8} p_i l_i$$

$$= (0.22)(2) + (0.20)(2) + (0.18)(3) + (0.15)(3) + (0.10)(3) + (0.08)(4) + (0.05)(5) + (0.02)(5)$$

$$= 2.8 \text{ binits/message-symbol}$$

The entropy H(S) is given by

$$H(S) = \sum_{i=1}^{8} p_i \log \frac{1}{p_i}$$

$$= 0.22 \log \frac{1}{0.22} + 0.20 \log \frac{1}{0.20} + 0.18 \log \frac{1}{0.18} + 0.15 \log \frac{1}{0.15} + 0.10 \log \frac{1}{0.10} + 0.08 \log \frac{1}{0.08} + 0.05 \log \frac{1}{0.05} + 0.02 \log \frac{1}{0.02}$$

$$= 2.7535 \text{ bits/message-symbol}$$

$$\therefore \text{Code efficiency } \eta_c = \frac{H(S)}{L} = \frac{2.7535}{2.8} = \mathbf{98.34\%}$$