Data Compression – Problem Set 1

(48) 1.

$s_1 = 1111$	$f_1 = .40$
$s_2 = 1110$	$f_2 = .15$
$s_3 = 10$	$f_3 = .10$
$s_4 = 01$	$f_4 = .09$
$s_5 = 1101$	$f_5 = .09$
$s_6 = 1100$	$f_6 = .09$
$s_7 = 00$	$f_7 = .08$

Sampling shows that when files from a certain source are parsed by $s_1, \ldots, s_7$, at left, then each s_j occurs with relative frequency approximately f_j (also at left) in the resulting source string.

Find the encoding schemes for replacing $s_1, \ldots, s_7$ obtained by the methods of (a) Shannon;
(b) Fano;
and (c) Huffman.

(d) Also, compute the compression ratio achieved by each of these methods, on files from the certain source referred to above, and compute the Shannon bound on the compression ratio, in these circumstances.

40
(38) 2. $S = \{0,1\}^2 = \{s_1, s_2, s_3, s_4\}$, where $s_1 = 00$, $s_2 = 01$, $s_3 = 10$, and $s_4 = 11$. These occur with relative frequencies $f_1 = .6$, $f_2 = f_3 = .15$, $f_4 = .1$. [From which you can infer that the original file has a lot of zeroes, compared to ones.]

10
[9] (a) Find the encoding scheme and the compression ratio arrived at by applying Huffman's algorithm to S, with these source frequencies.

20
[18] (b) Assume that each $s_i s_j \in S^2$ occurs with relative frequency $f_i f_j$, in the source stream. [So, for instance, 0001 occurs among consecutive four-bit strings in the original file with relative frequency $f_1 f_2 = (.6)(.15) = .09$]

Find the encoding scheme and the compression ratio achieved by applying Huffman's algorithm to S^2.
[Remarks: ① The encoding scheme here will have 16 lines.
② The assumption about the ~~real~~ relative frequency of $s_i s_j$ being given by the product $f_i f_j$ is a common simplifying assumption, in the absence of any real statistics about the relative frequencies of the "digrams" $s_i s_j$; please do not come to believe that this assumption is necessarily valid, or even necessarily a good approximation of reality.]

10
[~~9~~] (c) Compute the source entropy and the Shannon bound on the compression ratio.

12
(~~10~~) 3. $s_1 = 0100$
$s_2 = 0101$
$s_3 = 001$
$s_4 = 000$
$s_5 = 110$
$s_6 = 101$
$s_7 = 111$
$s_8 = 100$
$s_9 = ?$

The set $S = \{s_1, \ldots, s_9\}$ has the Strong Parsing Property. Find s_9.
Also, find all possible "leaves" (left-overs) of binary files parsed, left-to-right, by S.