CSEP 590
Assignment 1
Due Thursday, October 4, 2007

1. The Kraft-McMillan inequality

$$\sum_{i=1}^{n} 2^{-l_i} \leq 1.$$

provides a condition for the existence of a prefix code for n symbols where symbol a_i has length l_i. Design a *recursive* algorithm that constructs for an input sequence of positive integers $l_1, l_2, \ldots, l_n$ an output binary tree with leaves labeled $a_1, a_2, \ldots, a_n$ and the depth of the leaf labeled a_i is l_i. (Hint: there is a construction described on pages 33-34 of Sayood which may be helpful. However, it may be more useful to look at slides 53 and 54 of lecture 1.)

Your algorithm should start with n nodes thought of as a forest of n trees each with one node. In each recursive call, the quantity $s = \sum_{i=1}^{n} l_i$ is reduced by at least one. Generally, there are three cases to consider if there is a recursive call. If there are at least two symbols then either $l_1 = l_2$ or $l_1 > l_2$. The third case is when there is just one symbol left and $s > 0$.

Demonstrate your algorithm for the sequence $4, 4, 3, 3, 2, 2$.

2. In this problem we consider how to build Huffman codes for one symbol contexts. Consider the four symbols $\{a, b, c, d\}$ with conditional probabilities $P(x, y)$ defined by the table.

P	a	b	c	d
a	.1	.1	.1	.7
b	.4	.4	.1	.1
c	.1	.7	.1	.1
d	.2	.3	.2	.3

$P(x, y)$ is the conditional probability that the symbol immediately following symbol x is symbol y. The x is indicated by a row and the y by a column. Thus, each row must sum to 1. Define $P^*(x)$ as the probability of x in the long run. It can be seen, for example, that the probability of the symbol a is

$$P^*(a) = .1P^*(a) + .4P^*(b) + .1P^*(c) + .2P^*(d)$$

which is the sum for each symbol of the probability of the symbol times the conditional probability that the next symbol is a.

(a) Calculate the probability of each symbol. That is, calculate $P^*(x)$ for each x. To do this you need to solve a system of linear equations.

(b) Construct the Huffman trees for these contexts. (See slides 68-71 of Lecture 1)

(c) Calculate the bit rate of these trees as a code. This can be done using $P^*(x)$ for $x \in \{a, b, c, d\}$ and the bit rate for each Huffman tree.

3. There are applications where we need a prefix code for an infinite set of symbols such as the non-negative numbers $\{0, 1, 2, \ldots\}$. One way to do this is with the Elias γ-code defined as follows. In the γ-code, the integer $n \geq 0$ is coded by first writing $n + 1$ in binary, then preceeding it with m 0's where $m + 1$ is the number of bits that were just written. The γ-codes are started in the table:

number	code
0	1
1	010
2	011
3	00100
4	00101
5	00110
6	00111
7	0001000
8	0001001
⋮	⋮

(a) Explain why the γ-code is uniquely decodable.

(b) Encode the sequence 5 0 10 20 using the γ-code.

(c) Decode the sequence 100011011000010001 using the γ-code.

(d) Give an expression for the length of the γ-code of n as a function of n. Your expression should be fully general, but can have cases depending on how n relates to a power of 2.