CSC 2508S
Fall, 2002

October 2, 2002
DUE: October 21, 2002

Information: Quantification, Representation and Manipulation

Assignment #2
<**RELEASE #2**>

1. Arithmetic Coding [20 marks]

Either by hand or using a program, carry out the tasks below involving *Arithmetic Coding*. (Show your intermediate work by writing down, or printing out from your program, the intermediate values obtained at each stage of the computations.)

Your approach may count on the fact that all messages to be encoded or decoded will be nine or fewer characters in length. Consequently, there is no need to deal with shifting of bit string representations of the numbers.

Assume a source alphabet of four symbols (A, B, C, and D) with respective *independent* probabilities of occurrence (.4, .3, .2, and .1).

(a). Encode the string "DACABABACAD" producing a real number in [0,1).

(b). Decode the value .517353856 to produce a string of length nine composed of the four symbols.

2. Lempel-Ziv-Welch Coding [40 marks]

Write a program to do encoding and decoding according to the simplified LZW approach. Your dictionary implementation may be based on any of (i) text strings (easier, but less general), (ii) linked lists representing strings (simple and efficient), or (iii) a TRIE representation of the set of strings (complex, but interesting!). Your program should read input from a file that has the following format:

line 1:	D	% the number of symbols in the alphabet
line 2:	$S_1, S_2, S_3, \ldots, S_D$	% the symbols
line 3:	ENCODE or DECODE	% the operation to be carried out

The remainder of the file will be, for ENCODE, a string of characters over the alphabet **in which all spaces should be ignored**, or, for DECODE, a sequence of integers (separated by spaces) that represents an encoded text string. Conversely, the output for ENCODE should be a sequence of integers, and the output for DECODE should be a sequence of characters from the alphabet **(either with or without spaces between adjacent pairs of characters).**

Apply your program to each of the (four to six) files available for this purpose on the course web page.

For each of the six text files that you process, keep track of and display:

(i). the number of dictionary entries at the end, and

(ii) the ratio of the number of characters in the unencoded message to the number of integers in the encoded message.

<**See other side.**>

3. Calculating Entropy Rates Based on Context [40 marks]

Write a program to calculate an estimate of the *Entropy Rate* of an arbitrary text file by creating Markov models of **orders 0, 1, 2, 3 and 4. Note that the higher order models will provide the better estimates of the entropy rate, but also that the models of orders 3 and 4 will only be feasible when the size of the symbol alphabet is sufficiently small.**

If you find it difficult to base your program on a single scan of the file, you may use a first pass to determine the symbol alphabet, and a second pass to gather the relevant probabilities. **Such as initial pass can also be used to strip out all characters in the file that are *not* among the set of characters chosen to be the symbol alphabet.**

The larger the number of distinct symbols that appear in the file, the lower the order of dependencies you should try to handle. For example, if a file contains more than one hundred distinct characters, then you should attempt to obtain at most O(1) and O(2) dependencies; but if the file contains only 10 or fewer distinct characters, then it should be possible to all address O(3) and maybe O(4) dependencies. To explicitly control the number of characters considered to be in the symbol alphabet, you may simply ignore all occurrences of symbols that you don't wish to be considered as part of the symbol alphabet.

Once the transition probabilities for various amounts of context are found, your program should use them to produce estimates of the entropy rate of the document.

(a). Apply your program to the four test files for this question available on the course web page, and give an estimate of the entropy rate for each one.

(b). Next, apply your program to at least three other text files (of diverse natures) found in your directories (e.g., mail, latex, html, data or system files), and report your estimates of their entropy rates.

(c). Briefly discuss your findings from the experiments carried out in part (b).

Regarding Teamwork

For questions 1 and 2, I would like everyone to work on their own. For question 3, you may, if you wish, form pairs for the *writing* of the program. However, for the experimentation and explanation, each of you should again work alone.