Massachusetts Institute of Technology
Department of Electrical Engineering and Computer Science
Department of Mechanical Engineering

6.050J/2.110J **Information and Entropy** **Spring 2004**

Issued: February 9, 2004 **Problem Set 2** **Due: February 13, 2004**

Problem 1: It's in the Genes

DNA consists of a sequence of *codons*, each having three nucleotides (any of the four nucleotides, U, G, C, or T). Consider a 500-codon-long sample of DNA and the corresponding protein of 500 amino acids. You wonder about the information associated with this sample of DNA and the information associated with the resulting protein.

a. How many bits do you need to specify a single codon?

b. If you represent the information as a sequence of 500 codons, how many bits do you need to encode the sample of DNA?

c. There are twenty amino acids. How may bits do you need to specify an amino acid?

d. If you represent the information as a sequence of 500 amino acids, how many bits do you need to encode the protein?

e. Note that the number of bits you need is different using the two approaches. If you were designing a system to store the information needed to manufacture a protein, which approach would you use? Discuss the advantages and disadvantages of the two.

f. You have the opportunity to design a new life form which uses the same set of nucleotides and the same set of 20 amino acids, but uses longer codons. Your first design decision is how many nucleotides to include in one codon. If you include only three, then you can specify only one amino acid per codon. If you include five, you can specify a sequence of two amino acids. Consider all cases of 11 or fewer nucleotides per codon, and calculate for each case the "information compression" which we will define here as one minus the ratio between the number of bits per amino acid for the new form of DNA and the number of bits per amino acid in Nature's original code, namely 6.

Problem 2: Pay per Bit

An important consideration in the design of codes is efficiency. In fixed-length codes like ASCII each symbol (in the case of text, each character is a symbol) is represented in the same number of bits. In variable-length codes the more frequently occurring symbols are assigned shorter codes and less frequently occurring symbols longer codes. If all symbols are equally likely there is no advantage to variable-length codes.

The Penny-Dreadful Book Company needs an efficient code suitable for everyday English text, one that is more efficient than ASCII (7 bits per character). The text they are interested in consists mainly of lower-case letters. Two of their engineers (one of whom, incidentally, was educated at Caltech, and the other at MIT) have each devised a code, and the company has asked you to advise them which to adopt.

In the first proposed code (designed by the Caltech graduate) lower-case letters are represented by 5-bit sequences. Since there are only 26 such letters, there are 6 unused sequences. One of these is assigned to

'space', and one to the control code 'ETX' signifying the end of the text. The code is defined so that the four unused sequences are all of the form 101xx where each x is either 0 or 1. The remaining characters are encoded using 9 bits each, starting with 101 so that they cannot be confused with the codes for lower-case letters.

In the second proposed code (designed by the MIT graduate), lower-case letters, space, ETX, and four common punctuation marks (period, comma, question mark, and exclamation point) are represented by 6-bit sequences, all of which start with 0. All other characters are represented by 7-bit sequences which start with 1.

a. You need to represent ETX and all the ASCII characters except the control characters. This includes 26 lower-case letters, 26 upper-case letters, 10 digits, space, and 32 punctuation marks. Can both codes do this? Explain your answer.

b. If either code can not do this, see if you can modify the design slightly to permit all the characters to be represented.

c. Compare the number of bits required for a typical paragraph with 300 lower-case letters, 50 spaces, 30 upper-case, 10 common punctuation marks, 20 uncommon punctuation marks, and one ETX for whichever of Penny-Dreadful's codes that work, as well as ASCII and the typical 8-bit code used in computers.

d. It is pointed out that there is also need for some specialized control characters, such as new-paragraph and new-chapter, and additional punctuation marks such as em-dash, en-dash, and opening and closing curly single and double quotes. Describe in words how you would propose extending the second proposed code to allow 12 additional characters.

e. **Extra Credit:** Implement an encoder and decoder for the MIT graduate's code. For simplicity only include the first seven letters, space, period, semicolon, and ETX. Apply it to the phrase "Deb begged; Gabe bagged a cab." Using MATLAB is recommended, but not required.

Turning in Your Solutions

Turn in this problem set by e-mailing whatever M-files and diaries you may have generated, along with your answers to any problem(s) not done using MATLAB, to 6.050-submit@mit.edu. You may do this either by attaching them to the e-mail as text files, or by pasting their content directly into the body of the e-mail (if you do this, please somehow indicate where each file begins and ends). Alternatively, you may turn in your solutions on paper in room 38-344. The deadline for submission is the same no matter which option you choose.

Your solutions are due 5:00 PM on Friday, February 13, 2004. Later that day solutions will be posted on the web.