Algorithms in the Real World
Assignment #3 Due Date: 19 October 00

Problem 1: (30pt)
Let $H = \left[A^T I_{n-k}\right]$ be the parity check matrix for a binary linear block code, as described in class. Prove that the Hamming distance between any two codewords is d, where $d \geq 2$, if and only if the columns of H are $(d-1)$-wise independent. (The columns of H are $(d-1)$-wise independent if and only if no column can be expressed as a sum of $d-2$ or fewer columns. For $d = 2$, this implies that no column is all zeroes. For $d = 3$, this implies that no two columns are identical.)

Problem 2: (10pt)
Describe the elements and the addition and multiplication operations of the Galois field $GF(2^4)$, and show that one of the elements is a generator for the multiplicative group by enumerating the powers of that element. (For this problem, it is not required that you enumerate the results of all possible pairwise multiplications and additions of field elements.)

Problem 3: (10pt)
Suppose that there is a very inexpensive PCI board that implements an $RS(255, 223)$ Reed Solomon encoder and decoder in hardware. (This is most likely true!) The board encodes or decodes sequences of 223 bytes and can correct up to 16 errors in a sequence. You would like to use Reed-Solomon codes to protect your data against errors as it is transmitted over a wireless communication channel. Unfortunately, your radio experiments show that, at your transmission rate, bursts of errors tend to be longer than 16 bytes. Using the $RS(255, 223)$-encoder/decoder as a building block, design a system that can correct up to 64 consecutive errors in an 892-byte message.

Problem 4: (10pt)
Suppose that you are given the $RS(225, 223)$ hardware described above, but your message size is always 156-bytes, and your requirement for correcting errors is only 10 bytes rather than 16. For each message, you are willing to send 176 bytes of data and parity, but no more. Show how to build a system to transmit your messages using one $RS(255, 223)$-encoder and one $RS(255, 223)$-decoder.

Problem 5: (10pt)
Reed-Solomon codes can be used to correct "erasures" as well as errors. An erasure occurs when a symbol is known to be invalid in a particular position. As an example, a disk drive may realize that it cannot read data from a specific sector on a disk. How many erasures can be corrected using an $RS(n, n-k)$ code, assuming that there are no other errors? You do not have to give an efficient algorithm for correcting the erasures, but you must prove that there is sufficient information in the correct symbols to recover the symbols that were erased.