Mathematics 392 – Seminar in Coding Theory
Individual Problem Set 1
January 18, 2002

Due: Friday, January 25

In class, we introduced the following simple encoding scheme for binary strings abc of length $m = 3$: For each such abc we encode

$$abc \mapsto abc(a+b)(a+c)(b+c) \tag{1}$$

where the sums are taken *mod 2*. For example,

$$110 \mapsto 110011.$$

The resulting set of 8 words of length 6 is the code C:

$$C = \{000000, 001011, 010101, 011110, 100110, 101011, 110011, 111000\}$$

Let $w \neq w'$ be any two words in C, and let $B = \{x : x$ is obtained from w by changing any one position $\}$, and similarly $B' = \{x : x$ is obtained from w' by changing any one position $\}$.

A) Show that $B \cap B' = \emptyset$ for all w, w' directly, by computing all the sets B and comparing them.
B) The approach in A is not very elegant. Give a different proof that $B \cap B' = \emptyset$ by considering the way the codewords in C are formed using (1) above.
C) Is C as big as possible for a code with $n = 6$ keeping the property that $B \cap B' = \emptyset$ for all $w \neq w'$? Could you add another word of length 6 (or even more than one) to C and preserve this property? If so, find the largest possible code containing C with this property; if not, say why not.
D) Consider binary words of length n, and let w be any fixed word. How many words can be obtained by changing any ≤ 1 position in w? How many words can be obtained by changing any ≤ 2 positions in w? In general, how many words can be obtained by changing any $\leq k$ positions in w (where $k \leq n$)?

From the text: Read section 1.9 (we will not cover that section in class) and do problem 1.9.7 parts a, h