Due Monday Mar 1st in class. Each problem has 10 points.

Homework 2

1. Suppose three users are sharing the same channel to communicate with their respective partners. Suppose they choose to use CDMA and ask you to design the codes (or chipping sequences) for them. Suppose they want to use chipping sequence of length 4. Give a set of codes for these 3 users without interference between them.

2. Reed Solomon code.

   (a) Use the following numerical example to show the pipeline of coding and decoding scheme. Suppose the sender wants to send a data of three numbers (3, 40, 21), i.e., \( k = 3 \). And the sender wants to encode it to \( n = 6 \) pieces of data. Show exactly what the sender and receiver do in this case. (10pts)

   (b) Suppose the sender wants to send \( k = 50 \) data pieces. If the channel has a probability of \( p = 0.6 \) that a piece of data gets lost during transmission, what is the average number of data in the codeword that the sender should send out? Or, in other words, what is the expected number of \( n \)? (10pts)

   (c) (Extra credit) In the description of the Reed Solomon code we use erasure channels. What if during transmission some numbers are received but corrupted during transmission? Give some ideas of what the receiver should do to correctly decode the original data. For this problem you can use the Internet (except online forums). (10pts)

3. The Hamming code \((7, 4)\) takes a string of length 4 and outputs a codeword of length 7. This is by taking the multiplication of the input string with the encoding matrix:

\[
H = \begin{pmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 \\
1 & 1 & 0 & 1 \\
1 & 0 & 1 & 1 \\
0 & 1 & 1 & 1
\end{pmatrix}
\]
For example, an input string \( x = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} \) will be encoded to

\[
Hx = \begin{pmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 \\
1 & 1 & 0 & 1 \\
1 & 0 & 1 & 1 \\
0 & 1 & 1 & 1
\end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}
\]

Here sum is in fact XOR (i.e., sum with mod 2). The question is, how many flip errors can be detected by this code? Why?